

UNIVERSITETET I OSLO  
Institutt for informatikk

# Forbedrings- potensialet i Altinn fra et brukerperspektiv

Masteroppgave  
for graden Master of Science

Ram Yoga

1. august 2005





# ABSTRACT

Altinn is a Norwegian portal for public reporting over the internet. The application can be seen as the “flagship” for how the use of IT is supposed to ease reporting to the government for businesses, organizations, and regular citizens in Norway. The Altinn platform is considered an important step on the way to around-the-clock governance and is the result of cooperation between the Norwegian Directorate for Taxes, Statistics Norway and The Brønnøysund Register Centre amongst others.

As a flagship Altinn has been a visible target and the system has endured a lot of criticism – especially for a frustrating use experience and the lack of ease of use. This thesis is an attempt at analyzing problems in the user experience and finding improvements for these problems. This is done from several angles; I look for improvements in findability, usability and accessibility.

I have conducted a heuristic evaluation and conducted a WAI preliminary review of the user interface. The heuristic evaluation revealed a range of points to improve. The thesis presents possible solutions for better finding and administration of forms; use of attachments; improvements in the help-system, error handling; and saving of forms. In addition there have been suggested improvements in the information architecture of the forms.

After analysis of the Altinn-system it is clear that the application does not comply with guidelines for web accessibility. Even basic guidelines are not met, and the system does therefore not pass the WCAG-test to receive even an “A”-rating. When the system fails at this level it means that many people will not be able to use Altinn. People with disabilities, such as blindness, will for example not be able to use the application. The thesis presents a comprehensive list of errors and possible solutions to the problems so that Altinn may comply with WCAG guidelines.



## SAMMENDRAG

Altinn er et felles nettsted for innlevering av offentlige skjemaer på internett, og kan anses som "flaggskipet" for hvordan bruk av IT skal lette skjemabyrden for næringslivet. Etter hvert er løsningen også blitt utvidet med tilbud for privatpersoner. Altinn-plattformen er ansett som et viktig skritt på veien mot døgnåpen forvaltning og er et samarbeid mellom blant annet Skattetaten, Statistisk sentralbyrå og Brønnøysundregistrene.

Som "flaggskip" er Altinn et synlig mål, og løsningen har blitt utsatt for mye kritikk – spesielt for mangel på brukervennlighet. Denne oppgaven forsøker å se nærmere på hva som kan forbedres slik at løsningen blir bedre for brukerne. Dette gjøres fra forskjellige innfallsvinkler. I denne oppgaven ser jeg på forbedringspotensialet i hvor lett det er å finne skjema, brukervennlighet, og til slutt hvor godt Altinn fungerer i forhold til universell utforming.

Det er blitt gjort en heuristisk evaluering og en WAI preliminary review av løsningen. Den heuristiske evalueringen avdekket en rekke forbedringspunkter. I oppgaven legges det frem løsningsforslag for forbedring av hvordan brukeren finner skjema; skjemaorganisering; og bruk av vedlegg. Videre ble det også funnet forbedringspunkter i navigasjon i skjema; hjelpesystemet; feilhåndtering; og lagring. I tillegg har jeg sett på forbedring av oppbyggingen og utformingen av skjema.

Etter gjennomgang og analyse viser det seg at Altinn-løsningen ikke tilfredsstiller krav til universell utforming og tilgjengelighet. Selv elementære krav tilfredsstilles ikke og løsningen klarer ikke å passere minstekravene for å få "A"-stempelet. At løsningen feiler her betyr at mange ikke kan bruke Altinn. Funksjonshemmede brukere, for eksempel blinde eller synshemmede vil ikke kunne benytte seg av løsningen. I rapporten legger jeg frem forslag for hva som må utbedres slik at Altinn tilfredsstiller krav til universell utforming.



# INNHold

ABSTRACT.....	III
SAMMENDRAG.....	V
INNHold.....	VII
FORORD.....	IX
1. INTRODUKSJON.....	1
1.1. Formålet med oppgaven.....	1
1.2. Bakgrunn.....	1
Hva er Altinn? .....	1
Skjemaveldet .....	2
eNorge 2005.....	2
1.3. Problemstilling og avgrensning .....	3
1.4. Begrepsavklaring interaksjonsdesign/ informasjonsarkitektur .....	5
1.5. Overblikk over oppgaven.....	7
2. OM ALTINN .....	9
2.1. Hva er Altinn?.....	9
Altinns aner: Avgiverløsningen.....	10
2.2. Hvem bruker Altinn? .....	12
Etatene.....	12
Sluttbrukerne.....	12
Nettleserstøtte.....	13
2.3. Brukergrensesnittet i Altinn.....	15
Bakgrunn .....	15
Prinsipper som er brukt i utformingen av brukergrensesnittet til Altinn .....	16
Brukergrensesnittet i dag.....	17
2.4. Løsningsarkitektur for Altinn .....	20
Overblikk over løsningen .....	20
Applikasjonsarkitektur.....	22
2.5. Oppsummering .....	29
3. BEGRENSENINGER OG PROBLEMER I ALTINN.....	31
3.1. Introduksjon .....	31
3.2. Problemer med Forprosjekt Altinn .....	31
3.3. Kritikken mot Altinn .....	32
3.4. Begrensninger som følge av arkitekturen.....	34
Skjemamotor og -verktøy.....	34
Begrensninger som følge av batch-oppsett.....	35
Oppgaveregisterets innvirkning .....	35
Begrensninger i preutfylling .....	36
3.5. Oppsummering .....	37
4. ORGANISERING AV ALTINN.....	39
4.1. Introduksjon .....	39
4.2. Definisjon av organisasjons- / virksomhetsarkitektur .....	39
4.3. Motstridende interesser .....	39
Staten .....	40
Etatene.....	40

Leverandører .....	41
Brukerne .....	41
4.4. Organisering .....	42
Overordnet organisasjonsarkitektur .....	42
Organisasjonsarkitektur for Altinn i Skatteetaten .....	43
Brukeropplevelsen i Altinn: et todelt ansvar .....	45
4.5. Prosesser .....	46
Arbeidsprosess for definering av skjema .....	46
4.6. Oppsummering .....	47
<b>5. FORBEDRINGSPOTENSIALET I WEBPORTALEN .....</b>	<b>49</b>
5.1. Introduksjon .....	49
5.2. Metode .....	49
Heuristisk evaluering .....	49
5.3. Resultater .....	57
Heuristisk evaluering .....	57
5.4. Diskusjon .....	64
Heuristisk evaluering .....	64
Analyse og løsningsforslag .....	67
5.5. Oppsummering .....	99
<b>6. FORBEDRINGSPOTENSIALET I SKJEMAUTFORMING .....</b>	<b>101</b>
6.1. Introduksjon .....	101
Retningslinjer for utforming av elektroniske skjema i Skattedirektoratet: eSkjema SKD .....	101
6.2. Definisjon .....	102
6.3. Metode .....	102
Arbeidet i eSkjema SKD .....	102
Analyse av skjema (innholdsanalyse) .....	103
Heuristisk evaluering .....	104
6.4. Resultater .....	104
Erfaringer med Altinn .....	104
Heuristisk evaluering .....	105
Analyse av skjema .....	105
6.5. Diskusjon .....	106
Analyse av skjema .....	106
Heuristisk evaluering .....	111
6.6. Oppsummering .....	115
<b>7. FORBEDRINGSPOTENSIALET I UNIVERSELL UTFORMING .....</b>	<b>117</b>
7.1. Introduksjon .....	117
Hva er universell utforming og tilgjengelighet? .....	117
Hvorfor er universell utforming viktig for offentlige nettsider? .....	118
Altinn-kontrakten .....	121
Stortingsmelding 40 .....	121
W3Cs retningslinjer .....	121
7.2. Metode .....	123
Utvalget .....	123
Grafiske nettlesere .....	124
Ikke-grafiske nettlesere .....	124
Automatiserte verktøy .....	124
Manuell testing .....	124
7.3. Resultater .....	125
Grafiske nettlesere .....	125



Ikke-grafiske nettlesere .....	125
7.4. Diskusjon .....	133
7.5. Oppsummering .....	144
<b>8. KONKLUSJON .....</b>	<b>145</b>
8.1. Introduksjon .....	145
<i>Forbedringer fra to sider</i> .....	145
8.2. Forbedringer i webportalen .....	145
<i>Findability</i> .....	145
<i>Navigasjon og oversikt</i> .....	146
<i>Håndtering av vedlegg</i> .....	146
<i>Lagring av brukers arbeid</i> .....	146
<i>Hjelp</i> .....	146
<i>Feilretting</i> .....	147
8.3. Forbedringer i skjemaforming .....	147
8.4. Universell utforming .....	148
<b>9. VIDERE ARBEID .....</b>	<b>149</b>
9.1. Veien videre .....	149
<i>Brukertesting og prototyping</i> .....	149
<i>Realitetssjekk og kost/nytte</i> .....	149
9.2. Hva vil fremtiden bringe? .....	150
<i>Samordning og nye tjenester</i> .....	150
<i>Informasjonsbehovet burde være utgangspunktet for skjema</i> .....	151
9.3. Til slutt .....	151
<b>REFERANSER .....</b>	<b>153</b>



# FORORD

Brukervennlige informasjonssystemer er noe jeg har brent for lenge. Jeg har vært interessert i, og jobbet med brukergrensesnittdesign og webutvikling i nesten 10 år. Men det var egentlig ikke før jeg leste Jakob Nielsens "Designing Web Usability" i 2000 at jeg skjønnte at brukervennlighet faktisk var noe man kunne *jobbe* med. Helt siden da, helt siden begynnelsen av informatikkstudiet, har jeg visst hvilket fagområde jeg ville skrive masteroppgave om. – Det jeg ikke visste var hvor mye arbeid som går ned i en slik oppgave, og hvor mye som faktisk blir liggende igjen på "klippebordet". Arbeidet med denne oppgaven har vært lærerikt og givende på mange måter.

Først og fremst må en stor takk gå til min veileder Ingvil Hovig. Uten henne hadde denne oppgaven aldri sett dagens lys. Hennes evne til å vise meg veien, hennes utrettelige optimisme og konstruktive kritikk har vært til uvurderlig hjelp i mitt arbeid med denne oppgaven. Ikke minst har hun vært en god samtalepartner!

Jeg må også rette en takk til mine medarbeidere i eSkjema SKD-prosjektet. De gav meg mye innsikt i hvordan skjemaprosessene og Altinn fungerer i det daglige. Særlig vil jeg takke Lasse Udjus, Tor Nygaard og Ketil Storvik for mange verdifulle og interessante diskusjoner. Jeg vil også takke alle som hjulpet meg med oppgaven på en eller annen måte – det er etter hvert blitt mange!

En spesiell takk må også gå til Mamma og Pappa. Deres kloke ord og støtte gjennom oppveksten har gitt meg en god rettesnor som har hjulpet meg dit jeg er i dag, og vil hjelpe meg videre.

Sist, men ikke minst; en meget spesiell takk til min kjære Ida. Hennes støtte og kjærlighet har hjulpet meg mer enn hun aner!

Oslo, juli 2005

*Ram Yoga*



# 1. INTRODUKSJON

## 1.1. Formålet med oppgaven

Valg av tema og problemstilling for oppgaven kan knyttes til mine egne frustrasjoner med den offentlige internett-portalen Altinn våren 2004. Denne våren skulle jeg fylle ut selvangivelsen for næringsdrivende på nett for første gang. Som praktiserende informatiker som har hatt interesse for, og jobbet med, brukskvalitet og webutvikling i flere år ble jeg skuffet av Altinn. Det offentliges nye, store satsning som skulle gjøre hverdagen lettere for næringsdrivende var frustrerende! – Det var ikke enkelt. Det var ikke logisk. Det var heller ikke elegant. Jeg var heller ikke den eneste som syntes dette!

Da muligheten for å skrive oppgave for Skatteetaten bød seg var ikke veien lang til velge å skrive om Altinn. Formålet med denne oppgaven er følgelig å se på forbedringer i *brukeropplevelsen* i Altinn.

Begrepet brukeropplevelse dekker mange aspekter ved en persons opplevelse av et system, og den er ofte subjektivt. Utgangspunktet har vært at det er rom for forbedringer i brukeropplevelsen av Altinn. I denne oppgaven har jeg valgt å se på tre områder av brukeropplevelsen: brukervennlighet, finnbarhet og tilgjengelighet.

## 1.2. Bakgrunn

### Hva er Altinn?

Altinn er et felles nettsted for innlevering av offentlige skjemaer på internett, og kan kanskje anses som "flaggskipet" for hvordan bruk av IT skal lette skjemabyrden for næringslivet (og etter hvert privatpersoner). Altinn-løsningen er ansett som "et viktig skritt på veien mot en døgnåpen elektronisk forvaltning" [www.altinn.no (om Altinn)].

Tanken bak Altinn er å samle skjemaene til det offentlige Norge på ett sted, og presentere det i et helhetlig og lettbrukelig system. Dette skal gjøre det lettere for folk å finne frem til skjema. I og med at alt presenteres i samme system, vil brukerne heller ikke måtte forholde seg til flere brukergrensesnitt.

Altinn passerte i juli 2005 seks millioner innleverte elektroniske skjema og får stadig flere av det offentliges skjemaer. Samtidig har løsningen måttet tåle mye kritikk, spesielt for hvor vanskelig det er å bruke (både brukergrensesnitt og arbeidsflyt).

## Skjemaveldet

Bakgrunnen for Altinn kan i stor grad spores til "skjemaveldet" [Nygaard 2000: 4]. Dette er et uttrykk som viser til den store mengden med skjemaer som bedrifter og privatpersoner må fylle ut som del av rapporteringsplikten til det offentlige.

De største belasterne (i antall skjema) er Skatteetaten og Statistisk sentralbyrå. Men det er også et annet aspekt ved belastningen. Det er følelsen av at informasjonen man blir bedt om ikke føles nyttig. Rapporten om skjemabelastningen legger ofte vekt på at et skjema som utfylleren ikke ser nytten av vil føles mer belastende enn et skjema som utfyller skjønner nytten av, og ikke minst ser egennytten av.

Flere rapporter viser at bedriftene føler at de bruker mye tid til å rapportere til offentlige instanser [Nygaard 2000, 2001a, 2003]. Som et ledd i forenklingspolitikken har regjeringen som mål å gjøre omfanget av rapporteringspliktene til bedriftene mindre. I tillegg er det som del av IT-politikken vedtatt at *IT skal brukes til å forenkle rapporteringen gjennom bruk av elektroniske løsninger*. IT er altså ett (av flere) vedtatte midler for å nå målet om en enklere hverdag for bedrifts-Norge. Samtidig skal også offentlig sektor forenkles og effektiviseres.

## eNorge 2005

eNorge 2005 er regjeringens overordnede IT-politikk for perioden 2002 – 2005. Selv om moderniseringsarbeidet av etatene har pågått i lengre tid, er eNorge 2005 sentralt for det politiske støtteapparatet bak den nåværende omveltningen, og effektiviseringen av offentlig sektor. IT blir ansett som et viktig virkemiddel for å nå de politiske målene om et enklere Norge.

Regjeringen har i eNorge 2005 satt tre overordnede mål for IT-politikken [NHD 2002]:

- **Verdiskapning i næringslivet**  
*Utvikling og bruk av informasjonsteknologi skal bidra til verdiskapning gjennom økt innovasjon og konkurransekraft i norsk næringsliv.*
- **Effektivitet og kvalitet i offentlig sektor**  
*Informasjonsteknologi skal brukes til å effektivisere offentlig sektor og tilby nye og bedre tjenester til brukerne.*
- **Deltakelse og identitet**  
*Alle skal kunne utnytte informasjonsteknologiens muligheter, og IT skal bidra til å bevare og videreutvikle vår kulturarv, identitet og våre språk.*

I denne oppgaven vil jeg fokusere på de to siste punktene, spesielt det å "tilby nye og bedre tjenester" og det at "Alle skal kunne utnytte informasjonsteknologiens muligheter [...]" (min utheving).

Brukersentrerte tjenester er ansett som viktig. Rapporten legger også opp til tilgjengelighetsaspekter og at *alle* skal ha tilgang til elektroniske tjenester.

### 1.3. Problemstilling og avgrensning

Jeg har valgt følgende problemstilling:

**Forbedringspotensialet i Altinn fra et brukerperspektiv**

- Realistiske og urealistiske forslag
- Fra papirskjema, via e-skjema til e-dialog



**Figur 1: Peter Morvilles User Experience Honeycomb**

Brukerperspektivet kan defineres på mange måter. Jeg har valgt å ta utgangspunkt i Peter Morvilles "User Experience Honeycomb" [Morville 2004b]. Morville setter sammen flere aspekter han mener påvirker brukeropplevelsen.

I denne oppgaven har jeg avgrenset problemet til å se på *tre* av disse aspektene ved brukeropplevelsen:

- *Usable*;  
Et nettsted må være anvendbart eller brukervennlig.
- *Findable*;  
Nettsteder må være navigerbare, og brukere må finne det de trenger for å gjøre det de skal.
- *Accessible*;  
Nettsteder må være tilgjengelig for alle, de må altså følge standarder for universell utforming.

Av tidshensyn og for at oppgaven ikke skulle bli for stor har jeg valgt å ikke ta med de resterende aspektene i denne oppgaven.

Det er to måter å bruke Altinn på: enten via fagsystem (for eksempel årsoppgjørprogramvare), eller gjennom webportalen. I denne oppgaven vil jeg se på bruk gjennom *webportalen*.

Det er mange innfallsvinkler til forbedringer av et datasystem som Altinn. Man kan se på det tekniske / arkitektoniske perspektivet, eller man kan se på det økonomiske perspektivet osv. Brukerperspektivet er kanskje en av de viktigste innfallsvinklene. Brukerne er en av de viktigste suksessfaktorene for en offentlig internettportal. Uten brukere som faktisk benytter tjenesten vil en slik løsning bare være et kostnadssug.

I denne oppgaven vil jeg se på hvordan Altinn kan videreutvikles for å nå potensialet i løsningen. For å gjøre det vil jeg se på hvordan løsningen kan utvikle seg i et lengre perspektiv.

"Realistiske og urealistiske forslag" henspeiler på at jeg vil frigjøre meg fra mange av de teknologiske og organisatoriske begrensningene som finnes i dagens Altinn (både organisasjon og arkitektur). Disse begrensningene holder løsningen tilbake, og det kan være vanskelig å fokusere på brukerbehovene når rammebetingelsene minner deg om hva som er mulig å ikke mulig. Derfor vil jeg komme med både realistiske og urealistiske forslag. Det er verdt å merke seg at mange av dagens begrensninger vil løses over tid, men fordi de ofte overstyrer behovene til brukerfokus vil jeg prøve å se forbi disse.

"Fra papirskjema, via e-skjema til e-dialog" spiller på hvordan utfyllingsprosessen er. Målet er å få til en dialog mellom det offentlige og borgerne når det kommer til rapporteringsplikter. I stedet for at borgerne ensidig svarer på alle standardspørsmålene som det offentlige stiller, skal e-dialog hjelpe begge parter til å håndtere kun nødvendig / interessant informasjon. [Se forøvrig ELMER-rapportene].

Jeg vil jobbe videre med det arbeidet som gjøres i Skatteetaten for elektroniske skjema. Skatteetaten har lenge jobbet med denne problematikken og har noen av de mest komplekse behovene i forhold til elektroniske skjema og krav til Altinn. Kort fortalt har Skatteetaten større (lengre) skjema, større *antall* skjema; mer komplekse regler; mange beregninger og avhengigheter mellom skjema og felter; kompleks rollehåndtering osv.

Med utgangspunkt i erfaringene som er gjort i Skatteetaten vil jeg se på behov for fremtiden som er kommet fram i prosjektet "eSkjema SKD". Videre vil jeg basere meg på erfaringer og tilbakemeldinger til Altinn som allerede er samlet inn, og se på noen av de mer "urealistiske" forslagene som inntil videre er forkastet pga. dagens og morgendagens begrensninger.

Forhåpentligvis vil jeg komme med innspill og ideer som på sikt kan implementeres i Altinn og bidra til å gjøre løsningen bedre for sluttbrukerne.



## 1.4. Begrepsavklaring

### interaksjonsdesign/informasjonsarkitektur

De som jobber med brukervennlighet kommer fra mange forskjellige disipliner: ingeniører, psykologer, medievitere, informatikere, bibliotekarer, grafikere osv.

Dette, kombinert med den økende populariteten til internett har gjort at stadig flere på en eller annen måte jobber med brukeropplevelser. Dette har også ført til at det har vokst frem mange begreper som tilsynelatende dekker det samme.

Begrepene "interaksjonsdesign" og "informasjonsarkitektur" blir ofte brukt om hverandre. Jeg vil her forsøke å klare opp i begrepsbruken og gi en definisjon på begrepene slik jeg bruker de i denne oppgaven.

Jenny Preece har den mest overordnede definisjonen av interaksjonsdesign [Preece, Rogers og Sharp 2002: 6]:

*"[Interaction Design is] designing interactive products to support people in their everyday and working lives."*

Rosenfeld og Morville bruker mye plass på å definere informasjonsarkitektur. De sier blant annet at informasjonsarkitektur er [Rosenfeld og Morville 2002: 4]:

*"The combination of organization, labeling, and navigation schemes within an information system."*

og:

*"The structural design of an information space to facilitate task completion and intuitive access to content."*

Videre definerer de interaksjonsdesign slik [ibid: 10]:

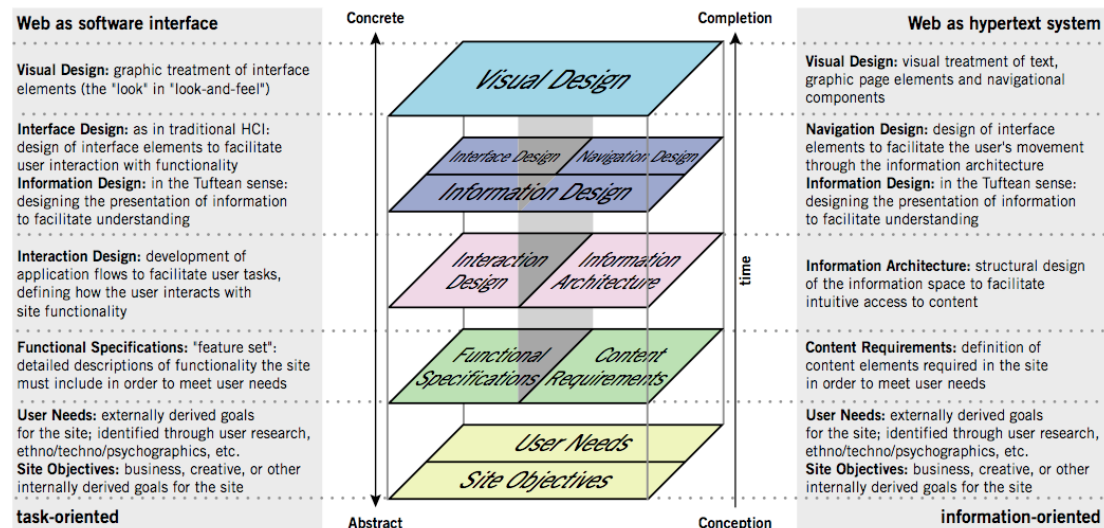
*"Interaction designers are concerned with the behavior of tasks and processes that users encounter in software and information systems at the interface level. [...] and are focused on helping users successfully achieve goals and complete tasks."*

Som vi ser er interaksjonsdesign og informasjonsarkitektur koplet sammen. Jesse James Garrett prøvde i 2000 å klare opp i disse begrepene. Han sier at disse to begrepene er to sider av samme sak. Grunnen til dette er webbens utvikling fra hypertekst-basert informasjonsrom til hypertekst-basert programvare. Han har sett på begrepene i internett/web-sammenheng og dualiteten i begrepene. Han skiller mellom web som *programvaregrensesnitt* og web som *hypertekstsystem* (illustrert i Figur 2) [Garrett 2003]:

*"Interaction Design: development of application flows to facilitate user tasks, defining how the user interacts with site functionality."*

Til sammenlikning er informasjonsarkitektur:

*"[...] structural design of the information space to facilitate intuitive access to content."*



Figur 2: elementene i bruksopplevelser, (c) Jesse James Garrett.

Steve Krug har den klart enkleste definisjonen [Krug 2000: 4]:

*"[...] interaction design (what happens next when the user clicks) and information architecture (how everything is organized)."*

I denne oppgaven vil jeg bruke begrepene slik:

*Interaksjonsdesign* dreier seg om hvordan den overordnede flyten i Altinn er. Hvordan brukeren henter, fyller ut og sender inn skjema. Det dreier seg også om på hvilken måte brukeren opererer systemet på – hva han klikker på.

*Informasjonsarkitektur* dreier seg om oppbygging og strukturering av selve skjemaene i Altinn. Hvilken rekkefølge innholdet er organisert i, hva som kommer på en side, hvilke spørsmål som hører sammen, og navngiving av grupper og navigasjonselementer.

### 1.5. Overblikk over oppgaven

Oppgaven starter i kapittel 2 med å gi en innføring i hva Altinn er. Her forteller jeg om bakgrunnen for løsningen og hvem som bruker den – både fra etatssiden og fra brukernes side. Videre gir jeg en innføring i hvordan Altinn er bygget opp teknisk og hvordan brukergrensesnittet og navigasjonsflyt foregår i dag.

I kapittel 3 ser jeg på problemene i Altinn. Først tar jeg for meg noe av kritikken som har kommet mot portalen. Deretter ser jeg nærmere på hvordan den tekniske løsningen påvirker brukeropplevelsen.

I kapittel 4 forteller jeg om hvordan Altinn er organisert. Altinn involverer mange parter, noe som bidrar til kompleksiteten i systemet.

Etter denne innføringen kommer tre kapitler som tar for seg tre forskjellige aspekter ved brukeropplevelsen i Altinn. I kapittel 5 ser jeg på selve webportalen Altinn og hva som kan forbedres i rammeverket og interaksjonsdesignet i systemet.

I kapittel 6 ser jeg nærmere på hvordan informasjonsarkitekturen til et skjema kan forbedres. Kapittel 7 tar for seg forbedring av tilgjengelighet gjennom universell utforming.

Konklusjonen kommer i kapittel 8. Der forsøker jeg å samle trådene og summere opp de viktigste funnene fra oppgaven.

Jeg avslutter oppgaven i kapittel 9. Der peker jeg på ting jeg ikke fikk tid til å gjøre, og hva som bør gjøres videre. Jeg kommer også inn på annet som ikke kom med i denne oppgaven, men som jeg gjerne skulle ha fordypet meg i.



## 2. OM ALTINN

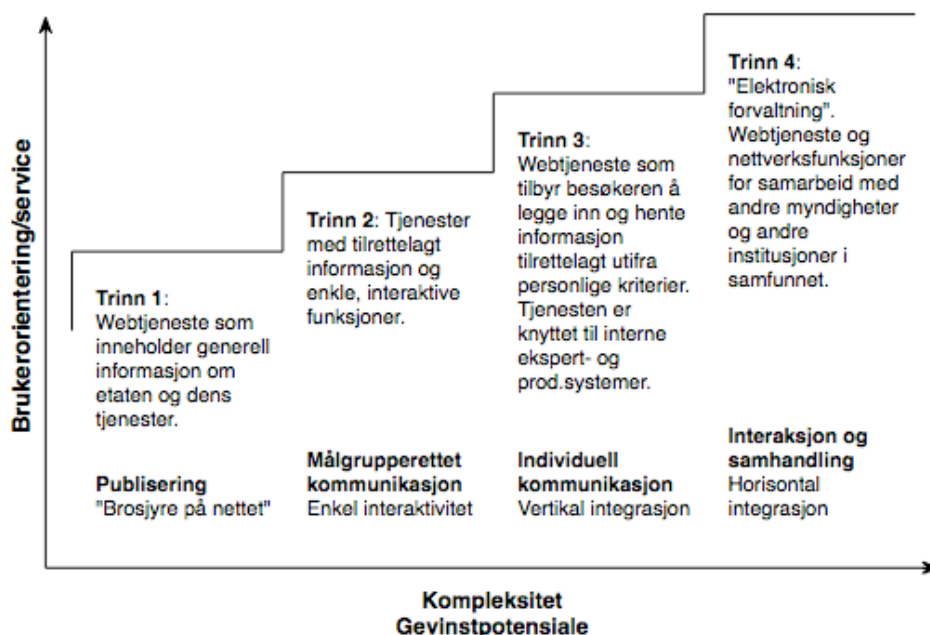
Dette kapittelet ser nærmere på Altinn. Først fortelles litt om bakgrunnen til løsningen, hvordan den ble til og hva som har skjedd med Altinn underveis. Det vil også bli sagt si litt om hvem som bruker systemet og hva slags tjenester som finnes i portalen. Mot slutten av kapittelet gis det en innføring i hvordan Altinn er bygget opp rent teknisk og det gis en introduksjon til brukergrensesnittet og navigasjonsflyt i løsningen.

### 2.1. Hva er Altinn?

Altinn er en offentlig portal, et felles nettsted for offentlige etater der du enkelt skal kunne rapportere informasjon til det offentlige. Altinn var *opprinnelig* en portal ment for næringslivet, men har etter hvert blitt utvidet med tjenester for privatpersoner – kanskje mest kjent er Skatteetatens elektroniske selvangivelse for privatpersoner som ble lansert i år.

Altinn er ansett som et viktig skritt på veien mot døgnåpen forvaltning og er på mange måter et pionér-arbeid med tanke på samordning og samarbeid mellom offentlige etater.

Offentlige internettjenester blir ofte vurdert etter hvor de ligger i den såkalte "tjenestetrappa" (se Figur 3). Denne figuren er en måte å vise hvilket modenhetsnivå en tjeneste ligger på.



Figur 3: Tjenestetrappa

Trappa blir forklart slik [AAD 2001: 13]:

*"Som det framgår av figuren, vil hvert trinn på denne trappen øke brukerservicen, men samtidig også kompleksiteten i tilretteleggingen av oppgaveløsningen. Gevinstpotensialet pr. enkelt-henvendelse øker for hvert trinn, men det stiller også større krav til elektroniske back-office løsninger (eks. systemer for elektronisk saksbehandling) og til informasjonsutvekslingen mellom virksomhetene."*

Vi kan si at Altinn ligger på nivå 4 i tjenestetrappa fordi systemet har nettverksfunksjoner og er integrert på tvers av flere etater. Denne horisontale integrasjonen er sentral nettopp for trinn fire i trappa.

Altinn startet som et samarbeid mellom Skattetaten, Statistisk sentralbyrå og Brønnøysundregistrene. Flere etater har sluttet seg til prosjektet i ettertid. Selve prosjektet ble startet opp høsten 2002 (etter et forprosjekt) og avsluttet i mai 2004, etter dette ble forvaltningen av Altinn lagt til Brønnøysundregistrene.

Tanken bak Altinn er å samle skjemaene til det offentlige Norge på ett sted, og presentere det i et helhetlig og lettbrukelig system. Dette skal gjøre det lettere for folk å finne frem til skjema. Fordi alt presenteres i samme system, vil brukerne heller ikke måtte forholde seg til flere brukergrensesnitt.

Formålet med løsningen er todelt. For *etatene* dreier det seg om at elektronisk innrapportering skal gjøre jobben deres enklere ved at data kommer rett inn til deres saksbehandlingssystemer uten at noen i etatene må "punche" data (dette faller nå rett på borgerne).

Formålet for *brukerne* (eller avgiverne som det heter i etatene) er å få redusert skjemabyrden og enklere kunne sende inn de data det offentlige trenger.

## **Altinns aner: Avgiverløsningen**

I 1997 gikk Brønnøysundregistrene (BR), Statistisk sentralbyrå (SSB) og Skattedirektoratet (SKD) sammen om utvikle den såkalte "Avgiverløsningen", som et ledd i koordinering av næringslivets innrapporteringsplikter til disse etatene.

Skattedirektoratet la sin del av dette arbeidet til SLN-prosjektet (System for likning av næringsdrivende). Her var det behov for elektronisk innrapportering siden saksbehandlingssystemene nå ble modernisert. Skattedirektoratet skulle bruke Avgiverløsningen til å ta imot likningsdata, SSB skulle bruke denne løsningen til å ta imot statistikkdata til sitt IDUN-prosjekt (Varehandel- og Industristatistikk), mens Brønnøysund skulle ta imot årsregnskap.

Det ble laget en felles kravspesifikasjon, som baserte seg på at alle skjema skulle etableres i Avgiverløsningen. I tillegg skulle det utvikles et grensesnitt mot fagsystemer for uttrekk av data. Dette var før utbredelsen av internett hadde kommet i gang for alvor, man utviklet derfor programvare for installasjon ute hos brukerne. Telenor ble valgt som leverandør, og løsningen baserte seg på Accelio

Jetform (Som i dag er kjøpt opp av Adobe) for skjema og Edifact-standard for meldingsformat, samt ulike alternativer for e-posthåndtering.

Skattedirektoratet fant tidlig ut at det var lite hensiktsmessig å legge mye ressurser i å utvikle likningsskjemaene. Det var to hovedgrunner til dette. Den ene grunnen fordi dette var veldig ressurskrevende. Det andre hadde sammenheng med at hovedvekten av innrapporteringene skjedde fra fagsystemer (årsoppgjørprogrammer). Etter hvert ble det derfor kun utviklet skjema for Brønnøysundregistrene og Statistisk sentralbyrå. Avgiverløsningen gikk i produksjon i 2002.

Under dette prosjektet, allerede i 2000, begynte samarbeidsgruppen (SKD, BR og SSB) å se på mulige løsninger for å håndtere brukere som fortsatt hadde behov for skjemautfylling. For denne brukergruppen ble det vurdert en enklere web-løsning. Til å begynne med snakket man om å etablere et helt uavhengig alternativ til Avgiversløsningen, og navnet "Altinn" ("alternativ innrapporteringskanal") ble født.

Etter hvert så man også nærmere på forvaltningen av avgiverløsningen og fant det problematisk å være programvareleverandør ut til markedet (både ut i fra distribusjonskostnader, drifts-/supportmessig og andre praktiske hensyn). Man ønsket derfor å flytte hele løsningen over på en web-basert plattform. I en periode så man på å utvikle en avgiver-komponent, som erstatning for avgiversløsningen og fagsystemkommunikasjonen, mot en Altinn web-løsning.

I 2000/2001 ble det nedsatt et forprosjekt for utredning av Altinn, som så på hele behovet rundt en full innrapporteringsløsning, inkludert også avgiver-komponenten. Arbeidet resulterte i kravspesifikasjon for Altinn (slutten av 2001), og derfra utviklet Altinn seg til den portalen den er i dag.

## 2.2. Hvem bruker Altinn?

### Etatene

Fra det offentliges side er det stadig flere etater og instanser som tilbyr skjema og tjenester gjennom Altinn.

Per mai 2005 er følgende etater med: Skatteetaten, Statistisk sentralbyrå, Brønnøysundregistrene, Lånekassen, Konkurransetilsynet, Kredittilsynet, Fiskeri- og kystdepartementet, Norges Bank, Økokrim og Produktregisteret.

Det er viktig for Altinn å få flest mulig etater til å tilby sine tjenester gjennom portalen for å ha dekning for profilen om en felles portal for innrapportering til det offentlige. Ikke minst er det viktig å få erfaringer innen elektronisk rapportering slik at feltet kan utvikle seg videre. I Norge er Altinn i så måte blant de viktigste aktørene, selv om enkelte andre etater har valgt andre løsninger.

Altinn bruker blant annet følgende innsalgsargumenter mot nye etater som vurderer å bli med i løsningen [[www.altinn.no](http://www.altinn.no)]:

- *Ferdig utviklet infrastruktur for elektronisk innrapportering som raskt kan tilpasses nye etater begrenser utviklingskostnader*
- *Felles portal for innrapportering med et felles grensesnitt på tvers av etater sikrer høy oppmerksomhet fra brukerne*
- *Gjennomtenkt og enkel integrasjon mot etatenes saksbehandlingssystemer*
- *Ferdigutviklet skjemaautviklingsverktøy inkludert i løsningen som utvikler elektroniske skjema basert på oppgaveregisterets metadatabase*

### Sluttbrukerne

Med så mange etater blir det stor bredde på sluttbrukerne av løsningen. Skatteetaten som helhet henvender seg til nesten hele befolkningen og har følgende brukere i alle ender av erfaringsskalaen.

Stadig flere bruker de elektroniske tjenestene til Skatteetaten. Over 1,8 millioner leverte selvangivelse for lønnstakere og pensjonister elektronisk i 2005. 70% av disse innleveringene var gjennom internett, det vil si gjennom Altinn [Skattedirektoratet 2005a].

Skatteetatens brukere av Altinn kan deles inn i to hovedgrupper: *private* og *næringsdrivende*. Altinn ble opprinnelig utviklet som en portal rettet mot næringsdrivende, men har med inntoget av selvangivelsen for private på internett begynt å gjøre seg gjeldende for privatpersoner også.

Når vi snakker om brukeropplevelse har vi et viktigere skille enn det mellom privat og næringsdrivende. Dette skillet er mellom *sporadiske brukere* og *profesjonelle brukere*.



Sporadiske brukere er personer som ikke bruker løsningen ofte. De fleste som er i kontakt med Altinn er sannsynligvis sporadiske brukere. Dette kan være vanlige lønsmottakere og eiere av enkeltpersonsforetak som bruker løsningen en til to ganger i året. De som rapporterer merverdiavgift bruker kanskje Altinn en håndfull ganger i løpet av året.

Profesjonelle brukere kan være såkalte service-agenter, som regnskapsførere og revisorer. Disse bruker Altinn i sitt profesjonelle virke og vil følgelig få et annet forhold til løsningen enn de sporadiske brukerne. Når det gjelder elektronisk innrapportering for næringsdrivende leveres 80 prosent av skjemaene inn via fagsystemer [Intern statistikk for SLN, Skattedirektoratet]. De resterende 20 prosent leveres via Altinn-portalen. Det er dermed naturlig å tro at de resterende som leverer næringslivsskjemaer via Altinn-portalen i høy grad er *sporadiske* brukere. Dette gjelder altså næringskjemaene – for de private vil nesten *alle* være sporadiske brukere.

Dette er viktig å innse når vi snakker om *hvem* som bruker Altinn til skjemautfylling. De sporadiske brukerne er inne så sjelden at de i essens *må lære seg tjenesten hver gang*. Brukere må finne frem på nytt hver gang. Dette stiller spesielle krav til løsningen og brukervennligheten.

### Nettleserstøtte

Nettleserteknologi går stadig fremover, men svært få brukere oppgraderer hele tiden. De som utvikler nettløsninger må derfor alltid ha nettleser-statistikken i bakhodet. For de aller fleste er det fortsatt slik at Explorer-familien er den som har størstedelen av besøkstallene. Det setter noen praktiske begrensninger på hva man kan utvikle av smarte løsninger og smidigere kode.

Altinn stiller ingen spesielle krav til maskinvare, men er optimalisert for 1024x768 skjermoppløsning. Netleser-støtten var ikke den beste de første månedene etter lansering, men dette er blitt gradvis utbedret. Løsningen skal nå støtte de fleste moderne nettlesere [www.altinn.no, mai 2005]. Under finnes en oversikt over nettlesere som støttes per mai 2005:

Windows	Linux	Mac
Internet Explorer 5.x (el. nyere)	(-)	(-)
Firefox 1.x	Firefox 1.x	Firefox 1.x
Mozilla 1.x	Mozilla 1.x	Mozilla 1.x
Netscape 7.x	Netscape 7.x	Netscape 7.x
Opera 7.x	Opera 7.x	Opera 7.x
(-)	(-)	Safari 1.x

Ut i fra tabellen kan vi se at det er mange moderne nettlesere som støttes. Basalkravet til løsningen er likevel Internet Explorer 5.x. 5.0-versjonen av Explorer har blitt en svært gammel nettleser, med dårlig støtte for webstandarder som HTML og CSS. Med dårlig støtte mener jeg at implementasjonen av webstandarder er mangelfull og tidvis feilaktig, slik implementasjonen var i de fleste nettlesere fra 90-tallet. Hvorfor

er dette viktig? For utviklere betyr dette at de må ta spesielle hensyn under koding slik at løsningen vil fungere i disse nettleserne. Eller som de sier i Web Standards Project [WASP 2002]:

*“The World Wide Web Consortium (W3C), along with other groups and standards bodies, has established technologies for creating and interpreting web-based content. [...]*

*Designing and building with these standards simplifies and lowers the cost of production, while delivering sites that are accessible to more people and more types of Internet devices. Sites developed along these lines will continue to function correctly as traditional desktop browsers evolve, and as new Internet devices come to market.”*

Explorer 5.5 har langt bedre standardstøtte enn 5.0, mens Explorer 6.0 som ble lansert i 2001 har relativt god støtte for standarder. Explorer 6 har likevel en del mangler [Festa 2003]. Microsoft har ikke lansert noen ny versjon av Explorer siden 2001 (Explorer 7 burde komme ut i betaversjon i løpet av høsten 2005). Da Explorer 6.0 ble lansert var det en svært moderne nettleser, med god standardstøtte etter daværende standard. Explorer har ikke blitt oppgradert på denne fronten siden 2001. I mellomtiden har konkurrenter som Firefox, Opera og Safari gjort mye for å bli gode på dette området.

### Nettleserstatistikk

For å få et lite innblikk i hva slags nettlesere som brukes nå har jeg hentet inn statistikk fra VG og Skatteetaten. Jeg har ikke hentet statistikk fra Altinn, fordi Altinn avviser nettlesere den ikke støtter.

Per mai 2005 hadde VG Nett følgende fordeling:

Internet Explorer	86%
Firefox	8%
Opera	5%

Til sammenlikning hadde VG nett i mai 2004 følgende fordeling: Internet Explorer 95%, Firefox 1,3% og Opera 2,8%.

For å få et publikum som ligger nærmere Altinn har jeg også hentet statistikk fra skatteetaten.no – det er naturlig at disse har nesten sammenfallende publikum siden Altinn har flere av Skatteetatens tjenester.

Per juni 2005 hadde skatteetaten.no følgende fordeling:

Internet Explorer	86,4%
Mozilla (inkludert Firefox)	5,1%
Opera	2,6%

Tallene fra året før tilsvarte tallene til VG.

Som vi ser representerer fortsatt Internet Explorer den desidert største brukergruppen. Selv om vi ser at stadig flere har valgt å bytte til andre nettlesere vil gamle versjoner av Explorer være med oss i mange år fremover. Det betyr at Altinn må fortsette å støtte (og ta hensyn til) nettlesere med manglende standardstøtte en god stund fremover.

Det vi ikke ser i denne oversikten er hva slags varianter av Explorer som er vanlige. Som nevnt er Internet Explorer 6.0 langt bedre enn 5.x-variantene. Dersom disse begynner å bli i mindretall vil man etter hvert kunne kutte støtten til disse. Da får man muligheten til å benytte flere moderne webstandarder og teknikker. Dette vil hjelpe på konsistensen mellom nettlesere og kan muliggjøre større fleksibilitet. Denne fleksibiliteten kan utviklerne bruke til å lage en bedre løsning av.

### 2.3. Brukergrensesnittet i Altinn

Brukergrensesnittet i Altinn er utviklet etter retningslinjer og overordnede prinsipper. Jeg vil her presentere bakgrunnen bak prinsippene og selve brukergrensesnittet i løsningen.

#### Bakgrunn

##### ELMER

Sommeren 1999 etablerte Nærings- og handelsdepartementet (NHD) en tverrfaglig kompetansegruppe som hadde som formål å fremskynde bruk av elektronisk rapportering i næringslivet. Denne referansegruppen bestod av flere enheter, deriblant de etatene som hadde flest skjemaer: Skattedirektoratet (SKD), Rikstrygdeverket (RTV) og Statistisk sentralbyrå (SSB), tre departementer og flere næringslivsorganisasjoner.

Denne kompetansegruppen dannet prosjektet ELMER (EnkLere og Mer Effektiv Rapportering) sommeren 2000. Prosjektet fulgte seks utvalgte bedrifter i ett år for å kartlegge rapporteringsplikter og teste ut løsninger som kunne gjøre rapportering enklere for næringslivet. Resultatet ble en rekke rapporter, pluss to eksempler på elektronisk rapportering: "elektronisk dialog" og "automatisk datafangst". Førstnevnte er et eksempel på hvordan skjema kan gjøres "på nett". Sistnevnte tar for seg hvordan bakenforliggende etatssystemer og næringslivets regnskapssystemer kan koples opp mot hverandre for automatisk (maskinell) rapportering. Disse rapportene inneholdt retningslinjer som senere ble brukt i utviklingen av Altinn-plattformen.

##### eSkjema-gruppa

I november 2000 dannet SSB, SKD og BR en kompetansegruppe for elektroniske skjema. Dette var en del av samarbeidet disse institusjonene allerede hadde om elektroniske innrapporteringssystem. SSB påtok seg hovedansvaret for dette samarbeidet. Gruppas viktigste oppgave var å komme med forslag til hvordan felles løsninger for elektroniske skjema kunne bygges opp og virke. Resultatene av dette

arbeidet ble publisert i Statistisk sentralbyrås håndbok nr. 81: "Råd for utvikling og utforming av webskjema" [Haraldsen 2004].

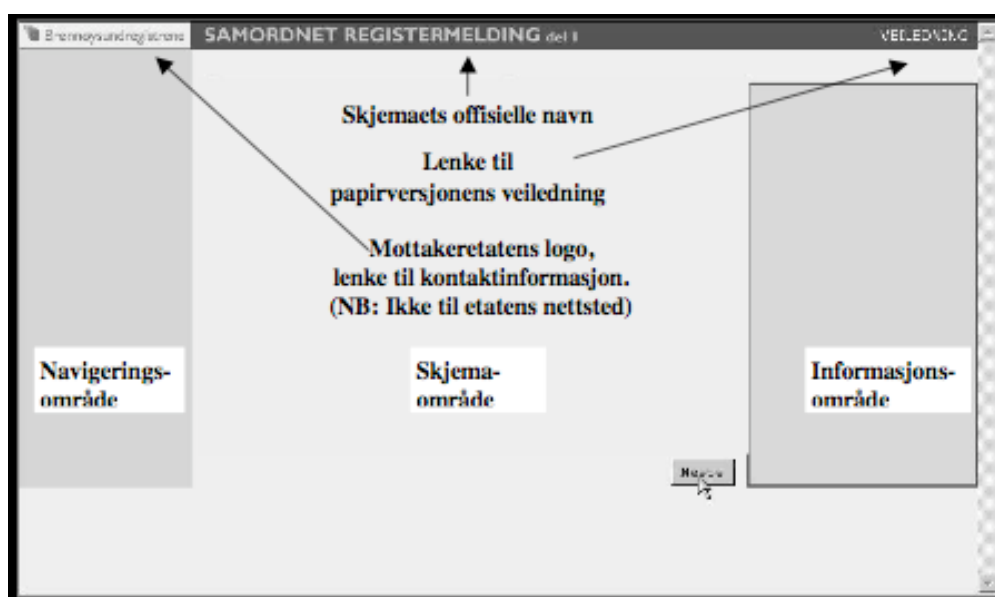
## Prinsipper som er brukt i utformingen av brukergrensesnittet til Altinn

### ELMER-prinsippene

De såkalte ELMER-prinsippene har vært viktige i utformingen av brukergrensesnittet til Altinn.

I [Nygaard 2001b] presenteres det en demo der ELMER-prinsippene illustreres. Det understrekes i rapporten at denne demoen *ikke* er utkastet til en mal, snarere tanker og forslag. – Det er *kimen* til en mal.

I dette kapittelet i rapporten belyser prosjektgruppa kravene til *elektronisk dialog* (se under), samtidig som de kommer med noen konkrete anbefalinger. Mange av disse anbefalingene er blitt brukt i Altinn. Skjerminndelingen til Altinn likner faktisk svært mye på demo-løsningen som ble fremstilt i forbindelse med ELMER. Figur 4 er tatt fra rapporten [Nygaard 2001b: 8] og viser inndelingen:



Figur 4: ELMER - Inndeling av områder på skjermen

Anbefalingene fra ELMER-prosjektet er for det meste ment å hjelpe på konsistens og gjenkjennelighet. Alle prinsippene er basert på tradisjonelle designprinsipper, som betyr at det er den *visuelle* designen det konsentreres om. Det sies ingenting om universell utforming, selv om flere av prinsippene understøtter de som har behov for tilpassede nettsider.

### Hva er en ”elektronisk dialog”?

I ELMER-rapporten ”Skjemaet som lærte å lytte – Eksempel på elektronisk dialog” legges det frem en liste over krav til det som kalles ”elektronisk dialog” [Nygaard 2001b]:

- *Elektronisk tilgjengelighet til skjemaet*  
(Hent og fyll ut på nett)
- *Adgang til skreddersøm av skjema varianter*  
(Innledende spørsmål snevrer inn antall spørsmål)
- *Adgang til selvvalgt forklaringsnivå*  
(Den som vil får veiledning)
- *Tilbakemelding ved feil og mangler under utfylling*
- *Mulighet for forslag til utfylling basert på allerede registrerte data*
- *Elektronisk videresending til relevante parter for digital underskrift*  
(Kan være aktuelt for styremedlemmer, advokater og revisorer osv.)
- *Enkel, elektronisk innsending*

Poenget med å bruke begrepet ”elektronisk dialog” er å skille det fra ”elektronisk skjema”. Et elektronisk skjema har vi når *”spørsmål og design er representert i maskinlesbar form og kan gjengis på en skjerm og/eller skrives ut på en printer”* [Nygaard 2001b: 4-5]. Det som skiller en dialog fra et vanlig skjema er først og fremst at spørsmål og forklaringer *tilpasses* den enkelte utfyller. En elektronisk dialog vil derfor skille seg fra dens papirbaserte variant; meningen er å utnytte de mulighetene elektroniske hjelpemidler gir og ikke bare kopiere papirskjemaet direkte til skjerm.

### eSkjema-gruppas retningslinjer

I [Haraldsen 2004] presenteres også en rekke retningslinjer for hvordan webskjema bør bygges opp. I likhet med ELMER-rapportene fokuseres det på høynivå prinsipper som avgjør hvordan selve applikasjonen skal fungere, f. eks. at brukeren når som helst skal kunne ta en pause og plukke opp arbeidet på en enkel måte.

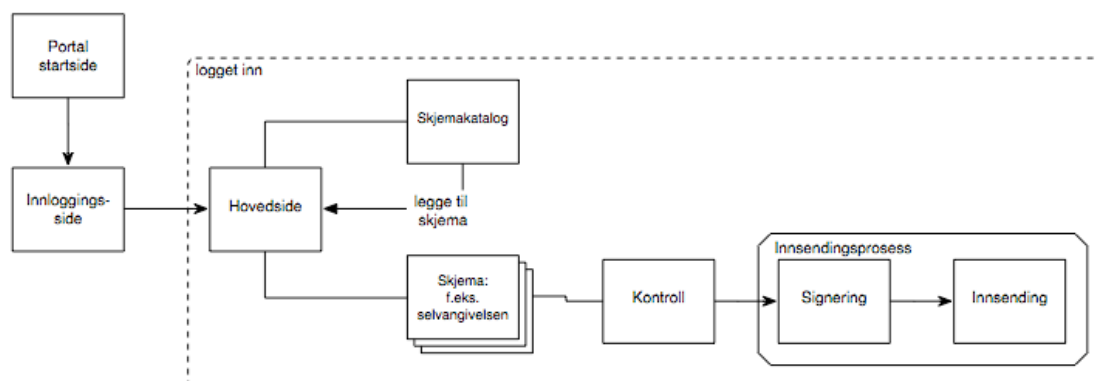
Disse retningslinjene tar heller ikke opp universell utforming eksplisitt. Men som med ELMER-prinsippene er det mange av retningslinjene som også mennesker med behov for tilrettelagte systemer vil dra nytte av.

### Brukergrensesnittet i dag

Altinn består i dag av en informasjonsportal som ikke krever innlogging. I informasjonsportalen finner man nyheter og informasjon om Altinn og skjema som finnes i portalen.

For å komme til selve skjemaløsningen kreves det at brukeren logger inn. I skjemaløsningen skjuler det seg en hovedside (”Hovedside” i Figur 5). Videre finnes det en skjema-katalog der brukeren kan finne skjema og legge dem til hans arbeidsliste. Fra arbeidslisten kan brukeren gå inn i selve skjemaet. Et skjema består av flere sider, og kan ha vedlegg. Vedleggene kan være andre skjema, filvedlegg og vedleggstabeller. I tillegg til dette finnes det en kontrollfunksjon som sjekker om alle

påkrevde felter er fylt ut m.m. Denne kan startes når som helst under utfylling, men må alltid kjøres før innsendingsprosessen begynner.



Figur 5: Overordnet navigasjonsflyt i dagens Altinn

Selve skjemasidene er delt opp i regioner. Regionene følger retningslinjene som er gitt i ELMER ganske nøye. I tillegg har løsningen blitt utvidet med mer funksjonalitet for å tilfredsstille kravene til Altinn.

I venstre kolonne ligger en navigasjonsmeny som viser hvor i skjemaet brukeren er og hvilke sider som finnes i skjemaet.

Figur 6: Brukergrensesnittet i Altinn i dag

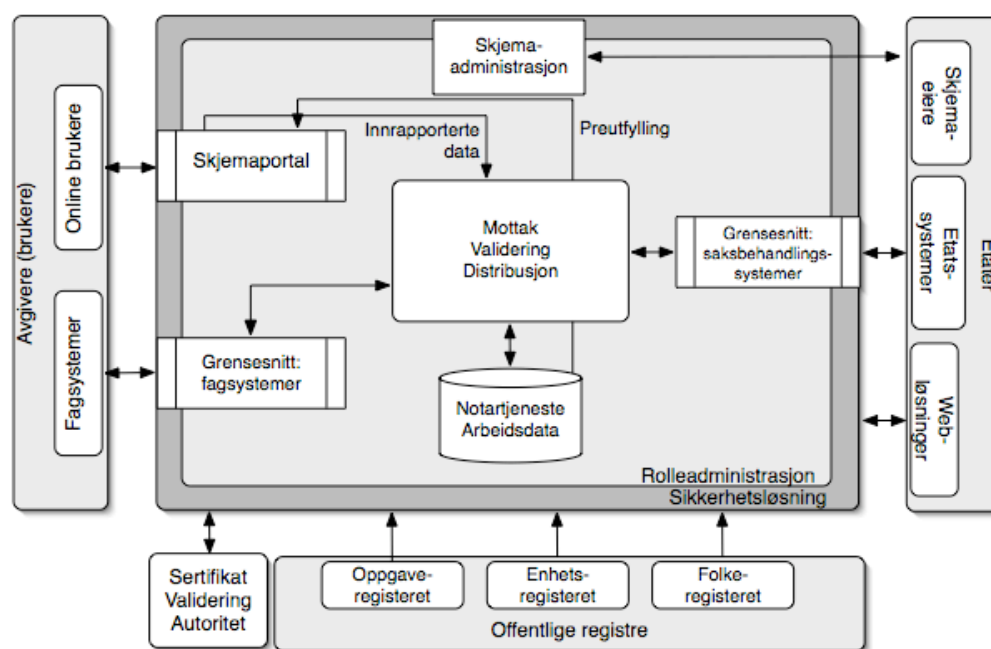
I midten av skjermbildet finner man selve skjemaet. Det er her brukeren fyller ut skjemasidene. Til høyre i bildet ligger et informasjonsområde. Dersom brukeren klikker på et av de små gule spørsmålstegnikonene i skjemaet kommer det opp hjelpetekst her. I tillegg vil feil komme opp når det kjøres skjema kontroll. Feilene kommer da i den røde fanen merket "Feilliste".

Nederst i skjermbildet finnes det enda et sett med navigasjonsknapper der brukeren kan bla fram og tilbake i skjemasidene, lagre skjemaet midlertidig, og for å starte kontroll av skjemaet.

## 2.4. Løsningsarkitektur for Altinn

### Overblikk over løsningen

For å få et overblikk over hvordan Altinn er bygget opp tar vi her en titt på hvordan den totale løsningen er satt opp.



Figur 7: Skisse av overordnet løsning (gjenskapet fra "Overordnet spesifikasjon" i Altinn-dokumentasjonen)

Innrapporteringsløsningen består av:

- En web-basert skjemaportal der brukere kan fylle ut, kontrollere og sende inn skjema. De har også tilgang til et arkiv der de kan se på tidligere innsendte skjema. Ansatte i etatene kan også se på skjema som de har tilgang til. *Det er først og fremst denne delen jeg vil fokusere på i denne oppgaven.*
- Grensesnitt mot fagsystemer (som årsoppgjørprogrammer) for mottak av helt eller delvis utfylte skjema. Det er her de fleste av innleveringene fra næringslivet i Altinn kommer inn.
- En formidlingssentral som mottar, validerer og distribuerer data. Formidlingssentralen har en arbeidsdatabase slik at brukeren kan lagre skjema for senere utfylling. Sentralen inneholder også en notartjeneste (det tidligere nevnte arkivet) som er en juridisk logg over innsendte skjema.
- Grensesnitt mot etatenes saksbehandlingssystemer. Gjennom disse grensesnittene kan etatene sende preutfyllingsdata og data om oppgaveplikt (kalt abonnement) og motta innrapporterte data.
- Grensesnitt mot offentlige registre. Gjennom disse grensesnittene mottas preutfyllingsdata, data om oppgaveplikt og data knyttet til faste roller som for eksempel daglig leder og styremedlemmer.



- Eventuelle grensesnitt mot andre webløsninger hos etatene som skal benytte tjenester fra Altinn.

Løsningen for etablering og endring av skjema består av:

- Grensesnitt mot Oppgaveregisteret (OR) for å motta metadata om skjema.
- Skjemaadministrasjonsverktøy for å:
  - Etablere og vedlikeholde skjema basert på metadata fra OR.
  - Etablere og vedlikeholde skjema uten metadata fra OR.

Felles for løsningen:

Grensesnitt mot offentlige registre. Gjennom disse mottas data som benyttes i forbindelse med brukeradministrasjon, for eksempel roller som er registrert i Enhetsregisteret.

Rolleadministrasjon som støtter tildeling av roller og rettigheter. Dette gjelder både rettigheter for avgivere og for etatsbrukere. Avgiver og etatsbrukere kan kun endre roller som ikke kommer fra enhetsregisteret.

### **Sikkerhetsløsningen og tilgangskontroll**

Sikkerhetsløsningen skal hindre uautorisert tilgang til data. Altinn bruker flere sikkerhetsnivåer for å få til dette. Sikkerhetsnivå benyttes i to sammenhenger, en for *autentisering* (innlogging) og en for *signering*.

Informasjonen er gradert i ulike nivåer, fra 0 til 4. Den innloggingsmetoden brukeren benytter seg av regulerer tilgang til informasjon og data, som vi ser i tabellen under:

Innloggingsmetode	Informasjonsgradering				
	0	1	2	3	4
Statisk passord	✓	✓	✓		
Engangskoder	✓	✓	✓	✓	
Programvarebasert PKI, nivå 2 eller sterkere	✓	✓	✓	✓	
Smartkortbasert PKI, nivå 3 eller sterkere	✓	✓	✓	✓	✓

Altinn tilbyr fire ulike innloggingsmetoder:

- 1) Statisk passord
- 2) Engangskoder sendt i post (som for personlig selvangivelse)
- 3) Engangskode tilsendt via mobiltelefon (SMS)
- 4) Smartkortbasert PKI (BuyPass).

I Altinn benyttes engangskode som primær metode for å etablere brukeridentitet, men innloggingssekvensen er utformet på en måte som oppfordrer brukere til å benytte SMS-engangskode. Løsningen er forberedt for senere innføring av ytterligere alternative PKI-baserte løsninger.

Til sammen dekker dette følgende autentiserings- og signeringsnivåer:

Sikkerhetsløsning	Autentiseringsnivå	Signeringsnivå
Statisk passord	0 - 2	0 - 1
Engangskode	0 - 3	0 - 2
Nettbasert PKI	0 - 3	0 - 2
Smartkortbasert PKI	0 - 4	0 - 3

### Altinn er en batch-løsning

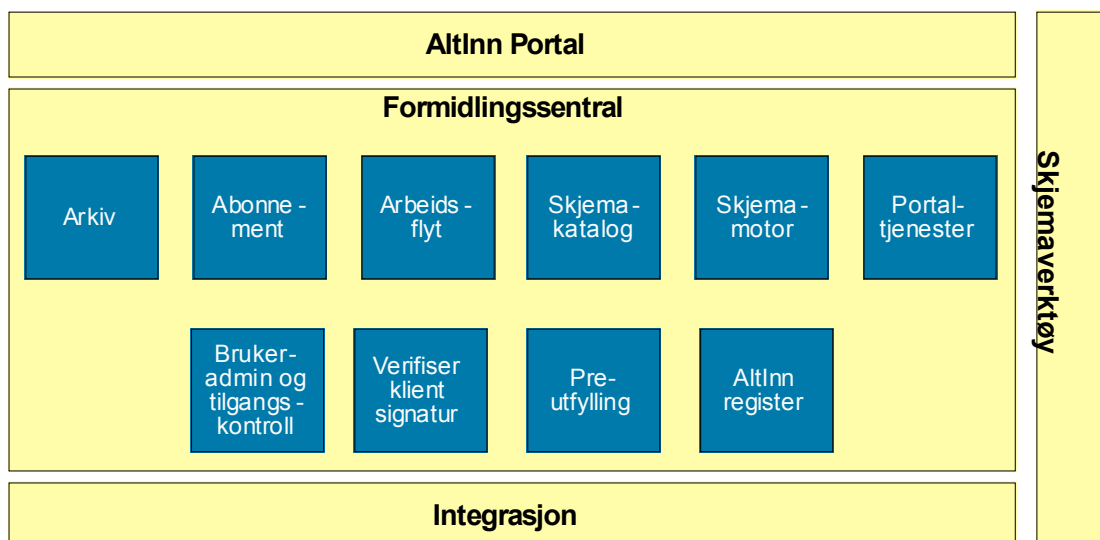
Det som ikke kommer frem på figuren over er at Altinn i dag er en *batch-orientert* løsning. Det betyr at endringer som rapporteres inn via Altinn ikke oppdateres i etatssystemer umiddelbart. Dette gjøres i bolker (engelsk: batch) i etterkant.

Hvorfor er så løsningen batch-basert? Dette skyldes først og fremst de bakenforliggende saksbehandlingssystemene og noen av registrene til etatene. Mange av disse bakenforliggende etatssystemene er gamle og er følgelig ikke bygget for å være koplet opp direkte mot andre systemer. Her snakker vi om problemer med tilkopling, ytelse og ikke minst problemer med hvordan systemer og registre skal oppdateres. Løsningen på disse problemene har vært å gjøre Altinn batch-basert med egne, lokale kopier av flere registre i Altinn. Altinn sender så inn data til etatssystemene ved gitte intervaller, og henter nye kopier av registrene. Fordelen med dette oppsettet er at Altinn ikke er avhengig av ytre systemer. Dette betyr at det er lettere å kontrollere ytelse og holde løsningen stabil – man trenger ikke bekymre seg for å holde koplinger til andre (gamle) systemer oppe m.m.

### Applikasjonsarkitektur

Her skal vi ta en nærmere titt på hvordan applikasjonen Altinn er bygget opp. Den består av fire hoveddeler:

- **Portalen**  
Her skjer utfylling og innsending. Dette er selve webgrensesnittet.
- **Formidlingssentralen**  
Formidlingssentralen håndterer innrapporteringsdata. Dette er den største delen av Altinn og inneholder mange underkomponenter. Underkomponentene blir gjennomgått senere i kapittelet.
- **Integrasjon**  
Her håndteres kontakt med andre systemer som fagsystemer, etatssystemer og registre.
- **Skjemaverktøy**  
Verktøy for å legge til og endre skjema. Her håndteres både metadata for Altinn-løsningen og den visuelle layouten av skjemaene i portalen.



**Figur 8: Oversikt over applikasjonsarkitekturen til Altinn (Tatt fra "Funksjonell spesifisering" i Altinn-dokumentasjonen)**

### **Altinn portal**

"Altinn portal" er selve nettportalen der brukere kan fylle ut og sende inn skjemaer for avgivere. "Avgiver" er et samlebegrep som brukes om både fysiske personer (privatpersoner) og juridiske enheter (som organisasjoner eller selskaper).

### **Kort om modulene i formidlingssentralen**

#### **Arkiv (notartjeneste)**

Arkivet inneholder en juridisk bindende logg over alle skjema som sendes inn. Dette er en viktig komponent som skal sikre at løsningen tilfredsstiller krav til saksbehandling.

Arkivet inneholder skjema brukerne tidligere har fylt ut og sendt inn (dette gjelder både skjema sendt inn via portalen og via fagsystem). Skjema i arkivet kan søkes frem og vises i portalen.

#### **Abonnement**

Abonnement gjør det mulig for Altinn å varsle og legge frem skjema som må fylles ut.

#### **Arbeidsflyt**

Arbeidsflyt-modulen skal holde oversikt over hvor i en arbeidsprosess et enkelt skjema er, for eksempel om det er under utfylling, eller avventer signering. Hvor i prosessen skjemaet er bestemmer hvor skjemaet kommer opp i løsningen og hvilke roller som får tilgang til skjemaet. Det er også denne modulen som håndterer mellomlagringen dersom en bruker fyller ut halve skjemaet for å fortsette utfyllingen en annen dag.

**Skjemakatalog**

Skjemakatalogen inneholder de forskjellige variantene av skjema som finnes i Altinn. Dette inkluderer kombinasjoner av skjema, versjoner, språkvarianter (alle skjema skal for eksempel finnes i både boksmåls- og nynorskutgave), og mediatype. Med mediatype menes format – Altinn har for eksempel både webskjema og PDF-utgaver tilgjengelig.

**Skjemamotor**

Skjemamotoren er den komponenten som styrer skjemautfyllingen. Motoren håndterer navigasjonsmenyen, utfyllingen av skjemaet (blant annet kalkuleringer), validering og presentasjon av hjelpetekst og feilmeldinger.

**Preutfylling**

Preutfyllingsmodulen skal hjelpe til med å forhåndsutfylle informasjon i skjema slik at brukerne slipper å fylle ut data som etatene eller offentlige registre allerede har.

Disse preutfyllingsdataene kommer fra offentlige registre (for eksempel Enhetsregisteret) og fra etatenes saksbehandlingssystemer.

**Altinn register**

Altinn er koplet opp mot offentlige registre og etatssystemer, men ikke for online oppslag. På grunn av at løsningen ikke har mulighet for direkte oppslag mot andre registre må løsningen ha egne kopier av offentlige registre og nødvendige data fra Enhetsregisteret, Folkeregisteret, poststedsregister, kommunetabell osv.

Data som mottas fra offentlige registre skal ikke inneholde sensitiv informasjon. Dette for å sikre at slik informasjon ikke benyttes ureglementert. For eksempel holder Altinns kopi av Folkeregisteret kun på fødsels- og personnummer, navn, postadresse og bostedsadresse.

**Integrasjon**

Integrasjonsdelen av Altinn har i oppgave å kommunisere med et utall av etatssystemer. De forskjellige etatssystemene er alle utviklet til forskjellige formål og i forskjellige tider. Det betyr at det er stort sprik i hvordan de er implementert og hvor gode tilkoplingsmuligheter de har.

**Mer om skjemamotor og skjemaverktøyet**

Når det gjelder brukeropplevelsen for utfylling av skjema i webportalen er det spesielt to deler som er sentrale; skjemamotoren og det tilhørende skjemaverktøyet. Skjemamotor og skjemaverktøy hører sammen som et tospann og er gjensidig avhengig av hverandre.

**Hva er en skjemamotor?**

En skjemamotor skal først og fremst håndtere skjema og skjemadata mot et presentasjonslag. I webportalen vil dette f.eks. være selve nettsidene med skjemaene. Her skal skjemamotoren håndtere logikken og forretningsreglene som gjelder skjemaene. Denne må også kunne presenteres til et utviklingsmiljø.

Dataene i skjemaet må kunne håndteres på en strukturert måte, slik at informasjonen kan lagres og bearbeides i en lengre arbeidsflyt. Skjemamotoren må også kunne presentere struktur og data på ulike formater, for eksempel HTML og PDF. I denne sammenhengen er det ikke nødvendigvis nok å kunne presentere én versjon av HTML. Det kan for eksempel være nødvendig å generere forskjellig HTML mot forskjellige nettlesere med og uten JavaScript.

En skjemamotor må i tillegg ha åpen arkitektur, støtte for standarder og være interoperabelt. At noe er interoperabelt betyr egentlig bare at det skal fungere mot flere systemer. I en verden der systemer skal kommunisere er det essensielt at de kan prate sammen. Dette gjøres ved å bruke etablerte standarder. Gjerne åpne standarder som er tilgjengelig for flere. Her kan vi snakke om tekniske standarder (som meldingsformater) og semantiske standarder. Semantiske standarder vil si felles beskrivelse av dataenes innhold og betydning (metadata) og standardisert presentasjon av disse.

Standard-støtte skal sikre kompatibilitet mot andre applikasjoner (og mot andre moduler i Altinn). At arkitekturen er åpen gjør det lettere å tilpasse/videreutvikle skjemamotoren der dette er nødvendig.

I tillegg til å jobbe mot et presentasjonslag må også en skjemamotor kunne lagre skjemadata i en database. Motoren må derfor tilby grensesnitt for tilgang og eksport av skjemadata.

For store systemer som Altinn er også referanser fra store installasjoner og modenhet i markedet viktig. I tillegg er det viktig at motoren kan passe inn i en større arkitektur. I utvelgelsen av produkt i Altinn-sammenheng var det også viktig at motoren hadde eget skjemautviklingsverktøy.

### **Skjemamotor i Altinn: Adobe Forms Server/Designer**

Skjemamotor/skjemaverktøy-kombinasjonen som er valgt i Altinn er Adobe Forms Designer med det tilhørende Adobe Forms Server. Versjonene som er brukt i Altinn er Forms Server/Designer 5.0 og varianter av 5.5 (patchet). Versjon 5.x av produktet er ikke utviklet av Adobe, men kjøpt av et annet firma (Accelio, tidligere Jetform) og pakketert som Adobe sitt eget. 5.x er et videreutviklet produkt basert på et gammelt designprogram som ble brukt for å lage papirblanketter. Adobe Forms Designer brukes altså både til å lage papirblanketter, og til å lage elektroniske skjema.

Produktet er skrevet for Microsoft plattform og er ikke integrert med Adobe sine andre programmer. Det er i svært liten grad kompatibel med Adobe sine øvrige formater, stort sett kan det skrive PDF og tildels lese PDF.

Altinn bruker fortsatt versjon 5.x. Men produktpakken har blitt oppgradert siden 5.x. Ved overgang til versjon 6.x ble produktet totalt reimplementert i Java. Man beholdt da noen av ideene i produktet, men ellers er resten endret. Nesten alle definisjoner, standarder, grensesnitt og integrasjonsmuligheter er endret. I versjon 7.x (som

kommer til neste år) blir produktet mer og mer integrert med Adobe sine øvrige produkter og standarder.

Skjemaverktøy / motor har stor innvirkning på brukergrensesnittet. Dette gjelder spesielt på kildekoden og oppbygningen av nettsidene. Fra Altinn-dokumentasjonen, [Accenture 2004b: 5]:

*"eGA vil være førende for bruken av skjemaverktøyet.*

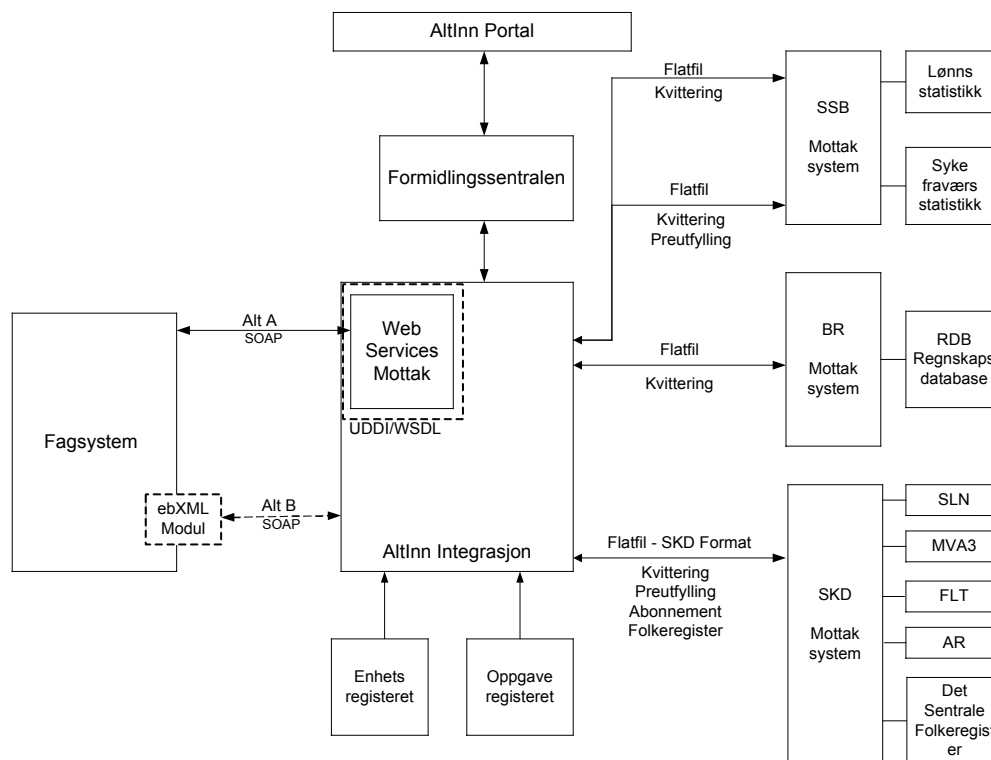
*Adobe Form Designer brukes for skjema utvikling, denne gir føringer for skjema utformingen og dermed brukeropplevelsen for sluttbrukerne.*

*ACA.net gir føringer for utformingen av Altinn Portalen."*

eGA (eGovernment Accelerator) refererer til Accentures produktpakke som er satt sammen av diverse hyllevare og spesialtilpasset for offentlig sektors elektroniske innrapporteringsbehov.

ACA.net (Avanade Connected Architecture for .NET) er et rammeverk som er bygget av Accenture og Microsoft for å støtte utvikling på Microsofts .Net-plattform.

## Litt om bakenforliggende systemer



**Figur 9: Overblikk over Altinn, med oversikt over noen av de bakenforliggende systemene i etatene. Tatt fra Fagsystem implementasjonsguide i Altinn-dokumentasjonen**

### Systemer i Skattedirektoratet

De data som blir innrapportert via Altinn må rutes til de riktige datasystemene i etatene for videre saksbehandling.

Direktoratet har flere bakenforliggende systemer for håndtering av data og saksbehandling. Det er her jobben blir gjort etter at data er blitt innrapportert. Jeg nevner noen av de viktigste her:

- System for Likning av Næringsdrivende (SLN)  
Dette er et relativt nytt system som brukes for saksbehandling av næringsdrivende.
- MVA3  
Dette er saksbehandlingssystemet for merverdiavgift.
- Forenklet Lønnsinnberetning og Trekkoppgjør (FLT)  
Dette systemet brukes for saksbehandling av innrapportering av lønn og skatt. Bedrifter/arbeidsgivere rapporterer inn lønnsdata for sine ansatte til dette systemet.
- Aksjonærregisteret (AR)  
Aksjonærregisteret holder rede på aksjer og eiere i selskaper og organisasjoner.

- Det sentrale folkeregister (DSF)  
Det sentrale folkeregister er den sentrale databasen for folkeregisteret og er således den base som skal være mest oppdatert. Tidligere har bedrifter og andre etater kunnet vedlikeholde sin egen folkeregisterdatabase for egne formål. Denne praksisen er endret til å gjøre online-oppslag mot det sentrale registeret.  
Folkeregisteret inneholder fødsels- og personopplysninger om alle norske statsborgere og er en viktig bestanddel i det som kalles "grunndata".  
Det ligger en rekke opplysninger i DSF, deriblant fødselsdato, personnummer, sivilstatus, adresse (post- og boadresse?), statsborgerskap, opplysninger om barn m.v.  
Altinn har en egen kopi av Folkeregisteret.
- Datastøttet Behandling (DSB)  
DSB er saksbehandlingssystemet for personlige skatteyttere. Dette er et distribuert system som finnes på de enkelte likningskontor. Dette systemet ble tilkopleet da full PSA (komplett elektroniske selvangivelse for personlige skatteyttere) ble lansert i Altinn.

### **Andre systemer**

Her vil jeg ta for meg systemer som ikke ligger hos SKD, men som likevel er viktige for Altinn. Dette kan enten være "system"-komponenter som leverer metadata (som oppgaveregisteret) eller det kan være registre (som Enhetsregisteret).

### **Oppgaveregisteret (OR)**

Oppgaveregisteret ble opprettet i 1997 med formål om å samordne og forenkle næringslivets oppgaveplikter til det offentlige. Hovedoppgaven er å holde løpende oversikt over oppgavepliktene mellom næringsliv og det offentlige med vekt på å hindre overflødig innsamling og registrering av opplysninger. Oppgaveregisteret har oversikt over hvilke opplysninger de ulike registrene og etatene krever fra de næringsdrivende og over nødvendige tillatelser for næringsdrift. Det er Brønnøysundregistrene som har ansvaret for OR.

Oppgaveregisteret har også en oversikt over datadefinisjoner (metadata). Det er disse datadefinisjonene som Altinn bruker for definering av skjema, preutfylling m.m.

Hvert dataelement blir tilordnet følgende egenskaper/attributter ved registrering i Oppgaveregisteret:

- Dataidentifikator
- Navn
- Definisjon
- Gruppe
- Type
- Kategori
- Relaterte datadefinisjoner



Disse egenskapene utgjør til sammen dataelementets datadefinisjon.

Oppgaveregisteret består av tre hovedsystemer:

- ORsys  
Internt system i Oppgaveregisteret som brukes til vedlikehold og administrasjon av oppgaveregisteret.
- ORetat  
Datamodelleringsverktøy som skjemamottakerne bruker.
- ORnett  
Tilgjengeliggjør datadefinisjonene på en webserver.

Oppgaveregisteret er et system som har stor innvirkning på Altinn-løsningen og er noe etatene må forholde seg til når de skal oppdatere datadefinisjoner og opprette skjemaer (både papir og elektronisk).

### **Enhetsregisteret**

Enhetsregisteret ble åpnet i 1995. Registerets oppgave er å samkjøre bedriftsopplysninger og å entydig identifisere organisasjoner og bedrifter i Norge.

Enhetsregisteret inneholder grunndata om enheter som har registreringsplikt i Arbeidsgiverregisteret, Merverdiavgiftsmantallet, Foretaksregisteret, Statistisk sentralbyrås bedriftsregister, Skattemanntallet for etterskuddspliktige eller fylkesmennenes registre over stiftelser.

## **2.5. Oppsummering**

Vi har i dette kapittelet sett på hva Altinn er. Vi har sett på bakgrunnen til systemet, hvordan det ble til, og vi har sett litt på hvordan løsningen fungerer. Videre har vi sett litt på hvem som bruker løsningen; både av etater og sluttbrukere. I tillegg har vi gått gjennom den tekniske løsningen på et overordnet nivå og sett litt på de forskjellige komponentene som Altinn består av, og hva slags systemer løsningen er koplet mot.

Vi har nå sett på ambisjonene til Altinn og hva det prøver å være. I kapittel 1 skal vi se nærmere på problemene i Altinn og hva slags begrensninger som finnes.



## 3. BEGRENSNINGER OG PROBLEMER I ALTINN

### 3.1. Introduksjon

Altinn er et stort og komplekst system. Dette kapitlet tar for seg en del av kritikken som er kommet mot systemet. Kapitlet vil også se på begrensningene som er gitt av den teknologiske løsningen, altså arkitekturen til systemet. Spesielt vil det bli sett nærmere på hvordan begrensningene påvirker brukeropplevelsen.

### 3.2. Problemer med Forprosjekt Altinn

Informasjonen her er blitt samlet gjennom diskusjon med ansatte i fagavdelingene i Skattedirektoratet.

Forprosjektet til Altinn var et IT-styrt prosjekt med overvekt av folk fra IT-avdelingen i Skattedirektoratet. Kun to stykker var med fra fagavdelingene, og ingen av disse var jurister. Konsekvensen av dette var at utfordringene til Altinn ble ansett å være av arkitektonisk art; det var integrasjonsproblematikk, arkitektur og PKI som var ansett som sentrale problemer.

På grunn av ovennevnte var det en rekke ting som ikke ble tatt hensyn til under forprosjektet og påfølgende kravspesifisering. IT-folkene hadde ikke gode nok kunnskaper om hvordan arbeidsprosesser til skattytere er. Spesielt ikke for næringsdrivende som gjerne setter ut selve føringen av regnskap og skatt til regnskapsførere. I tillegg til at selve arbeidsprosessene ikke ble tatt hensyn til var det også en del andre problemer:

- Noen selskaper har revisjonsplikt, andre ikke – her ble det antatt en forenklet modell.
- Hvem som signerer og hvordan signering foregår. Da fagavdelingene kom på banen til slutt ble det mye diskusjon om dette. Fagavdelingene var ikke fornøyd med løsningen fordi løsningsforslaget stred med loven. For Skattedirektoratet er det *signeringen* som er viktig, ikke hvem som fyller ut skjema.
- Personvern og taushetsplikt. Løsningen antok at det var subjektet selv som fylte ut skjema, men dette er feil. For næringsdrivende er det ofte profesjonelle som fyller ut, hvorpå daglig leder eller annen myndiggjort signerer de aktuelle skjema. Problemet med delegering av roller slik det var lagt opp i Altinn var at når man delegerte roller i løsningen fikk de tildelte tilgang til alt. Dette var fra fagavdelingene sett på som brudd på deres taushetsplikt, fordi uvedkommende da kunne få innsyn i sensitive data. Eksempelvis vil en revisor kunne få tilgang til sensitiv informasjon om din helsetilstand dersom du har dokumentert dette ifm selvangivelsen og din personlige inntekt. Men du har samtidig delegert rolle til regnskapsfører for å føre nødvendige tall til ditt eget enkeltpersonforetak.
- Notarfunksjonen var heller ikke tilfredsstillende i forhold til kravene i loven.

For fagavdelingene i Skattedirektoratet var det to ting som var sentrale i løsningen: signering og innholdet i skjemaene. Da løsningen gikk i produksjon var det fortsatt mye som ikke var godt nok i deres øyne. Dette har blitt jobbet med i ettertid. En del av disse manglene er blitt rettet, men det er fortsatt ting som må jobbes med fra deres ståsted.

### **3.3. Kritikken mot Altinn**

Da første versjon av Altinn offisielt ble lansert i februar 2004 varte det ikke lenge før løsningen ble utsatt for kritikk – både saklig og usaklig. Mange av brukerne var misfornøyde med løsningen og både leverandør, eiere og etater ble kritisert i media.

De fleste av feilene ble av Altinn-prosjektet og etatene kalt barnesykdommer som de regnet med å få utbedret etter hvert. Dette skapte reaksjoner blant brukerne og i media, men løsningen har gradvis blitt forbedret og fremstår i 2005 som en langt mer voksen utgave, uten mange av barnesykdommene.

Det følgende er en oppstilling av problemer med Altinn fra 2004. Kildene er mange: diverse artikler i Computerworld [Jahr og Tønnesen 2004a, 2004b, 2004c; Tønnesen 2004c, 2004a, 2004b, 2004d]; Digi [Ernes 2004a, 2004b]; tilbakemeldinger i diskusjonsforum på computerworld.no, digi.no, itavisen.no; og gjennom diskusjon med ansatte i Skattedirektoratet og Altinn sentralforvaltning.

Sentrale problemer med Altinn etter lansering inkluderte følgende:

- Komplisert pålogging;  
Mange hadde problemer med å komme inn i systemet for å begynne skjemautfylling.
- For høyt sikkerhetsnivå;  
Dette henger sammen med påloggingen. For mange brukere var det for vanskelig å komme inn, fordi kravene til sikkerhet var så store.
- Vanskelig å finne riktig skjema;  
Brukere hadde problemer med søkemotoren og det at mange skjema var "skjult" som vedleggsskjema.
- Rotete skjermbilder;  
Flere brukere oppfattet applikasjonen som uoversiktlig og vanskelig å bruke.
- Ingen automatisk summeringsfunksjon;  
Summering mellom felter fungerte ikke. Brukerne måtte derfor summere tall selv og taste inn svarene i summeringsfeltene manuelt.
- Dårlig med informasjon og hjelpemenyer;  
Mye av hjelpeinformasjonen ble ikke lagt inn til lansering grunnet stramme tidsfrister.
- Mye nedetid;  
Altinn var mye nede, spesielt mot slutten av innleveringsfrister da pågangen vokste kraftig.

#### **Problemer med innlogging**

Mange av brukerne hadde trøbbel med innloggingen. Fra etatenes side var det å sende ut pin-koder ansett som en sikkerhetsrisiko. På grunn av sikkerhetsbestemmelser ble det derfor i begynnelsen bare sendt ut fem koder om gangen, enten på selvangivelsen eller i eget brev. Skattedirektoratet bruker slike pin-koder til alle etatens tjenester.

Problemet med dette var at brukerne slapp opp for pin-koder veldig raskt. Slik løsningen var da den ble lansert måtte brukerne autentisere seg flere ganger under en sesjon, f.eks. når de skulle signere et skjema. Hvis de da ikke hadde flere pin-koder igjen ble de kastet ut av løsningen.

Det var også flere problemer med pin-kodene. De ble bare sendt ut et par ganger i uka, hvilket resulterte i at det kunne ta opp til to uker å få tilsendt nye pin-koder i de verste periodene med stor pågang.

Som en konsekvens av dette ble innloggingen i Altinn forenklet i slutten av 2004, slik at brukerne ikke ble bedt om passord like ofte som før. I tillegg har man endret praksis på engangskodene, slik at det i dag sendes ut 30 koder på hvert ark.

#### **Problemer med rollesystemet**

Rollesystemet til Altinn er bygget på Enhetsregisteret. Problemet med dette var at Enhetsregisteret på denne tiden ikke hadde spesielt god datakvalitet siden bedrifter og organisasjoner ikke var særlig flinke til å oppdatere denne informasjonen. Enhetsregisteret ble for det meste brukt i registreringsøyemed. Etter registrering var det mange selskaper som ikke oppdaterte informasjonen. Dette har selvfølgelig endret seg etter at registeret ble tatt i bruk Altinn, resultatet er at datakvaliteten har gått opp. Dette var forøvrig et av målene med å bake Enhetsregisteret inn i Altinn.

Trøbbelet med å bruke Enhetsregisteret som grunnlag for rolletildelingen i Altinn er at det ikke nødvendigvis er daglig leder som fyller ut skjemaene. Det er slik at daglig leder eller styremedlemmer skal signere skjema, men ofte er det andre som *fyller* ut skjemaene. Dette betydde at mange daglige ledere og styremedlemmer måtte logge seg inn i Altinn og delegerer tilgang til andre. Dette ble av mange av brukerne ansett som tungvint og unødvendig.

Andre problemer med å bruke enhetsregisteret er at ikke alle organisasjonsformer har organisasjonsnummer. Og det er også slik at ikke alle personer i Norge har fødselsnummer, det er enkelte som har såkalte D-nummer (fødselsnummer for utlendinger). De som har D-nummer kommer fortsatt ikke inn i løsningen, og må følgelig levere på papir.

#### **Skjema uten kalkyler**

Kalkyler (eller beregninger) ble vedtatt å kuttes rett før lansering. Dette betydde at brukerne ikke fikk noen summeringsfunksjoner – så de måtte selv legge sammen tall og plote de inn i sum-feltene. I tillegg måtte de flytte tall mellom forskjellige skjema manuelt.

Grunnen til dette var at kalkyler mellom underskjema ikke fungerte. Etatene mente at inkonsistensen (at kalkyler fungerte enkelte steder, men ikke andre steder) ville virke forvirrende. Det ble derfor vedtatt å fjerne alle kalkyler i skjemaene, inntil kvaliteten og konsistensen på kalkylene var god nok.

Kalkylene kom på plass i løpet av 2004. Dagens Altinn har både automatisk overføring av tall og summeringsfunksjoner på plass. Det er dog fortsatt enkelte steder tall må overføres manuelt mellom forskjellige skjema.

### **Signering/innsending**

Problemer med forståelsen av hvordan utfyllingsprosessen egentlig foregår resulterte i en merkelig situasjon der alle skjema måtte signeres to ganger, selv om avgiveren ikke hadde revisor. Først måtte daglig leder fylle ut skjemaet. Deretter måtte revisor signere (selv om organisasjonen ikke hadde revisorplikt). Dette resulterte i mange forvirrede brukere. SKD måtte legge beskjeder ved innsendingsknappen om at de uten revisor måtte plukke opp skjemaet igjen og sende inn nok en gang. Mens de som skulle bruke revisor måtte passe på å stoppe, slik at de *ikke* signerte på revisors vegne.

Problemet ble forverret ved at etatene kun hadde 1024 tegn på seg til å skrive denne teksten – inkludert HTML-kode.

## **3.4. Begrensninger som følge av arkitekturen**

### **Skjemamotor og –verktøy**

Kombinasjonene skjemamotor og skjemaverktøy setter til dels store begrensninger på flere forbedringsaspekter ved brukeropplevelsen.

Et av de største problemene med Adobe-pakken i versjon 5.x er at strukturen på skjemaene er flate og hvert element er uavhengig av de andre. Det gjør blant annet at det ikke er mulig å forholde seg direkte til Oppgaveregisteret sine definisjoner. At skjemastrukturen er flat at dataene ikke kan lagres strukturert.

Skjemamotoren håndterer heller ikke dynamiske tabeller, noe som har stått på ønskelisten til etatene en stund. I stedet må man bruke dagens vedleggstabeller.

Implementasjon av kalkylemotor mot lagrede data krever masse transformeringer av data. Dette skyldes igjen den flate skjemastrukturen – det er vanskeligere å flytte data mellom felter i et skjema og mellom skjema når det må gjøres mange beregninger for å finne ut hvor feltene finnes.

I nyere versjoner av produktpakka til Adobe (7.x og i en stor grad 6.x) benyttes det hierarki i skjemaoppsettene. I tillegg støtter programvaren seg i større og større grad på standarder for skjemadata, eksempelvis deler av XForms-standard. XForms er

W3Cs navn på deres nye skjemastandard. Denne standarden skal være plattformuavhengig og langt bedre til å bygge applikasjoner med.

Produktpakka produserer gyldig XML-kode, men standarden man har valgt (XFT og XFD) er svak (på grunn av nevnte flate skjemastruktur). HTML-koden som produseres fra denne XML'en er ikke gyldig, selv om XML'en er det. Årsaken er at det genereres ut svært mye ekstra informasjon (eksempelvis JavaScript) og det benyttes HTML tagging (bl.a. layers og positioning) som fungerer best med Microsoft sin nettleser. HTML-koden som produseres følger altså ikke HTML-standard.

Skjemamotorens flate struktur på skjemaene gjør det vanskelig å implementere kalkyler (beregninger) mellom underskjema; altså mellom hovedskjema og vedleggsskjema eller mellom vedleggsskjema.

Skjemamotoren setter også begrensninger på bruk av sporvalg og silingsspørsmål. Etatene har hatt ønske om å kunne legge inn sporvalg midt inne i skjemaene. Men skjemamotoren tillater kun sporvalg i begynnelsen av skjemaene.

Det er planer om å oppgradere skjemamotoren i Altinn til versjon 6 eller versjon 7 (når den kommer), men dette utsettes stadig. Hovedgrunnen til dette er at en oppgradering av skjemamotoren vil kreve at *alle* skjema i Altinn må bygges på nytt siden de nye versjonene av Forms Server bruker en annen (bedre) standard på oppbygningen av skjema.

### **Begrensninger som følge av batch-oppsett**

Det at Altinn er en batch-basert løsning har en del begrensninger. Først og fremst betyr det at regelhåndtering i saksbehandling må gjøres i ettertid – etter at dataene er blitt sendt til de bakenforliggende etatssystemene. Saksbehandlingssystemene til etatene har utførlige algoritmer for maskinell behandling og kontroll av innrapporterte data. Disse vil ikke være tilgjengelig i Altinn. Dersom dette skal gjøres i Altinn må de dupliseres – altså bygges på nytt i Altinn.

Fordi løsningen er batch-basert er det altså umulig å gjennomføre noen form for toveis (umiddelbar) kommunikasjon mellom Altinn og etatssystemer. Mange av kontrollene som finnes i etatssystemene kunne med fordel blitt tilgjengeliggjort i Altinn, siden det på mange måter nå er utfyller selv som gjør sin egen saksbehandling. For brukerne kan det være en fordel å få tilbakemeldinger om mangler og feil så snart som mulig, i stedet for å vente på svar fra etaten etter innlevering.

### **Oppgaveregisterets innvirkning**

Altinn er sterkt knyttet til Oppgaveregisterets metadatabase, siden skjemadefinisjonene må forholde seg til Oppgaveregisterets spesifikasjoner. Dette var et politisk valg som ble gjort tidlig. En av de viktigste grunnene var for å øke

kvaliteten på Oppgaveregisteret. Før Altinns tid var kvaliteten på databasen til Oppgaveregisteret lav fordi oppdateringer av databasen skjedde sporadisk. Ved å innlemme databasen i en løsning som Altinn har de etatene som bruker Altinn blitt nødt til å oppdatere databasen. Dermed har kvaliteten på databasen økt, og den oppfyller i dag langt mer av potensialet og hensikten bak opprettelsen av Oppgaveregisteret.

### Hjelpetekst

Oppgaveregisteret begrenser hjelpetekstene i skjemaene. Fordi hjelpetekstene er en del av skjemadefinisjonene som ligger i OR betyr det at det er OR som setter rammene for hva man kan gjøre med hjelpetekstene. Det at OR ikke inneholder presentasjonsdata betyr blant annet følgende:

- Ingen mulighet til formatering av tekst (fet skrift, eller kursiv)
- Ikke mulighet til å angi avsnitt eller overskrifter (man kan kun bruke linjeskift).
- Ingen mulighet for hypertekstlenker til andre hjelpetekster, eller andre steder på nettet, f.eks. Lovdata.

### Sortering av nedtrekkslister

I nedtrekkslister er det i dag ikke mulig å sortere valgene på en fornuftig måte, fordi man er avhengig av rekkefølgen som dataene ligger i OR. OR burde følgelig ha en måte å sortere disse valgene på – men sorteringen må være gjort på en måte som er fornuftig for brukeren. Det betyr f.eks. at det ikke alltid bør sorteres alfabetisk.

### Begrensninger i preutfylling

Preutfylling fungerer per i dag kun fra *ett* register om gangen. Det er derfor man for eksempel bare får ferdig utfylt fødselsnummer, *eller* organisasjonsnummer (se Figur 10).

The screenshot shows a web form titled 'Avgiver'. It contains the following fields and elements:

- Navn: Text input field.
- Adresse: Text input field.
- Postnr.: Text input field.
- Poststed: Text input field.
- Organisasjonsnr.: Text input field.
- Fødselsnr.: Text input field containing '123456789 10'.
- Henvendelse rettes til: Text input field.
- Telefonnr.: Text input field.
- Antall årsverk utført av ansatte: Text input field with a yellow question mark icon to its left.
- At the bottom, a question 'Har virksomheten skiftet næringsadresse i løpet av inntektsåret?' is followed by two radio buttons labeled 'Ja' and 'Nei'.

Figur 10: Preutfylling fungerer kun fra ett register om gangen.



Dette gjør at det i enkelte tilfeller må gjøres urimelige valg i etatene der de må velge hvilket register de skal preutfylle fra. For brukerne er det vanskelig å skjønne (og irriterende) hvorfor *noen* felter er preutfylte, mens andre ikke er det når etatene helt klart vet hva som skal inn der.

I Figur 10 ser vi et eksempel fra Altinn der en person skal registrere informasjon til selvangivelsen. Han er næringsdrivende og har et enkeltpersonforetak og skal i skjermdumpen fylle ut et vedleggsskjema til selvangivelsen om sitt firma. Siden selvangivelsen er personlig ligger skjemaene under hans personnummer og ikke under organisasjonsnummeret. Det betyr at han registrerer data som seg selv, derfor fylles fødselsnummeret ut automatisk. På grunn av begrensningene i preutfylling blir ikke organisasjonsnummer og foretakets adresse fylt ut automatisk.

## 3.5. Oppsummering

Dette kapittelet har tatt for seg noen av problemene i Altinn som stammer fra forprosjektet. Det er også blitt sett på kritikken som kom mot Altinn etter lansering i 2004, og hva slags problemer som påvirket brukerne.

Videre har kapittelet tatt opp noen av føringene som legges av den valgte teknologien og arkitekturen i Altinn. En del av disse føringene må man leve med i nærmeste fremtid, fordi de er så store deler av systemet at det ikke enkelt lar seg reparere. For eksempel vil utskifting (eller oppgradering) av skjemamotoren bety at man bytter ut store deler av Altinn, og det vil kreve konvertering av alle dagens skjema.



## 4. ORGANISERING AV ALTINN

### 4.1. Introduksjon

Kompleksiteten i Altinn skyldes ikke bare den tekniske infrastrukturen og systemarkitekturen. Organiseringen av Altinn er også med på å komplisere bildet. For å forstå denne kompleksiteten er det nødvendig å si noe om hvordan Altinn er organisert.

Dette kapittelet vil ta opp hvordan Altinn er organisert i forhold til ledelse, etater og utvikling. Det vil også berøre arbeidsprosesser, for eksempel hvordan skjema legges inn, for å videre illustrere systemet.

### 4.2. Definisjon av organisasjons-/virksomhetsarkitektur

En virksomhetsarkitektur kan sies å bestå av to deler [Christensen 2005]:

- Strukturen på organisasjonen/ virksomheten (den statiske delen)
- Prosesser, som arbeidsflyt, roller og rettigheter (den dynamiske delen)

I tradisjonell betydning ville en virksomhetsarkitektur vært innenfor én virksomhet eller organisasjon. Men fordi Altinn består av flere organisasjoner vil jeg her se på hvordan Altinn er organisert. Jeg vil først se på helheten, deretter på hvordan Skatteetaten er satt opp for å håndtere Altinn.

Fordi vi ikke ser på en tradisjonell virksomhet her, vil jeg referere til organiseringen som *organisasjonsarkitektur* og ikke virksomhetsarkitektur utover.

### 4.3. Motstridende interesser

Det er mange interessenter i en offentlig og tverretattlig løsning som Altinn. Disse interessentene har alle forskjellige behov og ønsker som tidvis er motstridende. Interessekonfliktene er viktige fordi partene vil påvirke løsningen for å få gjennomslag for nettopp sitt syn.

De primære aktørene er disse:

- Staten
- Etatene
- Leverandører
- Brukerne

Disse aktørene har forskjellige ståsteder og underroller, alt etter hva slags stilling, interesser man har osv. Jeg skal under ta for meg de forskjellige perspektivene og

prøve å identifisere hva som kjennetegner de forskjellige ståstedene og hva som motiverer de.

## Staten

Som alle andre er staten (som helhet) opptatt av å gjøre en effektiv jobb. Siden staten representerer hele folket vil det være naturlig å tro at mange av ønskene til staten sammenfaller med enkeltbrukernes ønsker. Vi ser at dette gjenspeiles i IT-strategien til regjeringen og eNorge-planen [NHD 2002]:

- Bruker i sentrum
- "Et enklere Norge"

Samtidig som staten vil gjøre tjenestene enkle og gode er staten også opptatt av at dette ikke skal koste samfunnet for mye. Det er tross alt staten som finansierer disse tjenestene. Det er ikke bare snakk om gjøre det enklere for at brukerne skal få en bedre hverdag, det er også snakk om effektivisering av statsapparatet og etatene. Ved å flytte mer av saksbehandlingen ut til brukerne, kan etatene (og dermed staten) spare penger. Et av målene med de nye elektroniske tjenestene er effektivisering [MinSide-prosjektet 2004]:

*"Bli din egen saksbehandler. Bankene har klart det – nå er det forvaltningens tur."*

Det som kanskje er mest særegent med staten er at statens politikk kan endres raskt. Den politikken som til en hver tid utøves kan endres ved et regjeringsskifte eller dersom det politiske klimaet skifter. Dette er kanskje ikke så aktuelt for det området vi nå diskuterer; bruk av IT for å effektivisere det offentlige og gjøre offentlige tjenester lettere og mer tilgjengelig for folk flest. Dette er det for det meste enighet om blant partiene – det er *hvordan* man skal gjøre dette det strides om.

## Etatene

Etatene er også opptatt av å gjøre en effektiv jobb. Ved å tilrettelegge tjenestene for brukerne er det håp om å øke kvaliteten på innrapporterte data, og at opplysninger kommer inn i tide.

Etatene er utsatt for stort effektiviseringspress fra staten. Dette er en av hovedgrunnene til at det satses på elektronisk rapportering og elektroniske tjenester. Ved å få data rett inn i etatssystemer kan store ressurser spares på "punching", ikke bare tidsmessig, men også kvalitetsmessig.

Etatene er selvfølgelig mest interessert i å verne om sine interesser, slik at de kan gjøre en best mulig jobb etter sine premisser.

Brukerorienteringen kommer som følge av krav til effektivisering fra staten, men også som følge av krav fra brukerne.

I etatenes øyne er det noe som er vesentlig forskjellig mellom det offentlige og det private; det offentlige har ikke kjøp og salg av tjenester. I det offentlige er det snakk om *myndighetsutøvelse*. Den nye trenden med brukerorientering fra det offentlige kan derfor virke unaturlig, selv om også etatene ser at brukerorientering er lurt, men da mer for å selv kunne gjøre en mer effektiv myndighetsutøvelse.

Fra Skatteetatens overordnede strategi 2005-2008 [Skatteetaten 2004]:

- *Vi skal gjøre det enkelt å handle riktig.*
- *Vi skal bekjempe svart økonomi og skattekriminalitet.*
- *Vi skal samarbeide aktivt med andre.*

For Skatteetaten ser vi at brukerorienteringen faller under det første punktet. Etaten skal tilrettelegge det slik at det blir enkelt for borgerne å handle riktig.

### **Leverandører**

Leverandørene er selskaper som skal tjene penger. Det betyr at de ikke bare er interesserte i å lage og tilby gode løsninger – de skal tjene penger mens de gjør det. I tillegg har mange leverandører valgt plattform og programvarepakker. Å holde seg til få plattformer/pakker gjør at leverandørene kan fordype seg i hva som er mulig å få til i den aktuelle løsningen og er også kostnadseffektivt for dem. Men dette gjør også at leverandører kan bli mer motvillige til å prøve alternative løsninger som de ikke er kjent med.

### **Brukerne**

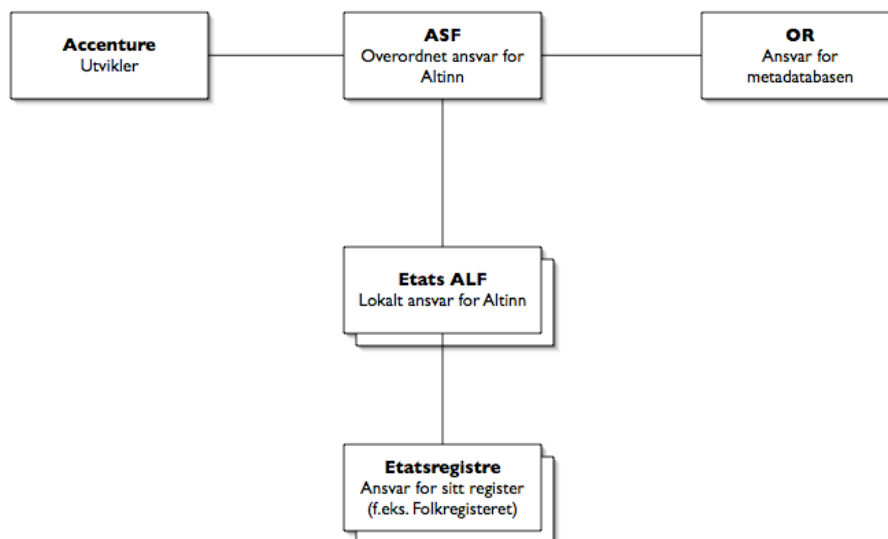
Brukerne (innbyggerne) vil ha enkle, tilgjengelige tjenester (uavhengig av tid og sted) som gir nytteverdi og som minker byrden på dem.

I dagens travle samfunn forventer brukerne å kunne gjort ting når det passer *dem*. Mange har ikke tid eller mulighet til å trappe opp på et offentlig kontor i kjernetiden. Samtidig er det blitt stadig høyere krav om å kunne få ytt tjenester og service på kundenes ( i dette tilfelle innbyggernes) premisser. Folk forventer i stadig større grad at det offentlige skal være serviceinnstilte.

Mange (staten inkludert) ser på IT som middelet til å modernisere det offentlige og gjøre samspillet mellom det offentlige og innbyggerne bedre. Gjennom gode IT-løsninger skal kravene til borgerne løses. Staten forventes å tilby moderne IT-baserte tjenester, gjerne på internett. Med kombinasjonen IT og travelheten forventer også stadig flere at saksbehandlingen skal gå raskere. Og de vil ha oppdatering om status på sin sak. Fra borgernes ståsted dreier det seg ikke om myndighetsutøvelse, men ytelse av service.

## 4.4. Organisering

### Overordnet organisasjonsarkitektur



Figur 11: Forenklet oversikt over Altinns organisasjon

#### Altinn sentralforvaltning (ASF)

Altinn sentralforvaltning er det koordinerende organet for Altinn. Avdelingen ligger under Brønnøysundregistrene, men er fristilt i forhold til resten av Registrene for å ivareta rollen som uavhengig koordinator og leder av Altinn.

ASF har flere arbeidsoppgaver:

- Koordinering og ledelse  
ASF skal fungere som et styrende organ mellom etater og leverandør. Det er ASF som skal ha et overordnet blikk og koordinere innspill til videreutvikling og forbedring.
- "Misjonsvirksomhet";  
En av oppgavene til ASF er å selge inn Altinn til nye etater.
- Opplæring  
Nye etater må ha opplæring i bruk av portalen og hvordan de administrerer og legger inn nye skjema.
- Ansvar for portal-delen av Altinn  
Det er ASF som vedlikeholder og videreutvikler informasjonsportalen i Altinn. Informasjonsportalen er den delen av Altinn som brukerne kan bruke før de logger inn.

#### Altinn lokalforvaltning (ALF)

Hver etat har en Altinn lokalforvaltning som skal være bindeleddet mellom ASF og etaten.

Etatenes ALF har varierende oppgaver, alt etter etatens størrelse. Men de fleste ALFer har følgende roller:

- Lokal koordinering
- Kommunikasjon med ASF
- Ansvar for etatens skjema i Altinn

#### Etatsregistre og Oppgaveregisteret

De etatene som har registre som brukes i Altinn er også med i lokalforvaltningene hos sin etat. Oppgaveregisteret (OR) er egentlig et lokalt register som er lagt under Brønnøysundregistrene og skal egentlig være ordnet under Brønnøysundregistrene sin ALF. Fordi Oppgaveregisteret er såpass sentral i Altinn er det ofte direkte kontakt mellom OR og Sentralforvaltningen.

#### Accenture

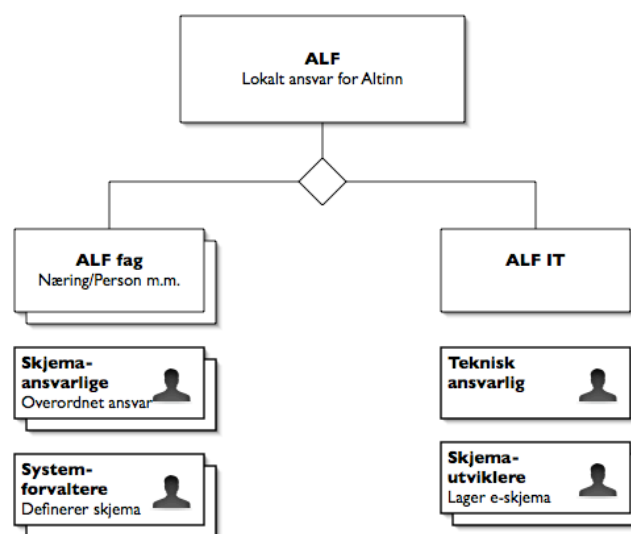
Accenture er (den nåværende) leverandøren av Altinn. Alle endringsforslag kommer til Accenture via ASF.

Accenture har følgende hovedoppgaver:

- Drifte produksjonsmiljøet (gjøres i dag av Ergo gjennom Accenture)
- Drifte utviklingsmiljøet
- Fikse bugs
- Videreutvikle løsningen

De har også kompetanse på løsning og skjemautvikling. De bidrar med konsulentbistand og support direkte mot de etatene som trenger det.

### Organisasjonsarkitektur for Altinn i Skatteetaten



Figur 12: Oversikt over Altinn lokalforvaltning i Skattedirektoratet

**Altinn lokalforvaltning (ALF)**

For en generell beskrivelse av ALF, se over. Skatteetatens ALF er delt inn i to hoveddeler: ALF fag og ALF IT. Disse representerer henholdsvis fagavdelingene (fastsetting næring, fastsetting person, merverdiavgift osv.) og utviklerne.

**Fagavdelinger**

Fagavdelingene har det faglige ansvaret for sitt område. Det er fagavdelingene som analyserer behovet for innrapporteringen innen sitt felt. Ved innrapporteringsbehov er det fagavdelingene som oppretter skjema i OR.

**Skjemaansvarlig/skjemaer**

I følge brukerdokumentasjonen for skjemaautvikling for Altinn [Accenture 2004a]:

*"Skjemaer bestemmer hvordan skjemaet skal se ut i form av hvilke data skal inn, hvilke tekster skal benyttes osv."*

**Systemforvaltere**

Skatteetaten har svært mange skjemaer. Per juli 2005 har Skatteetaten over 90 skjema i Altinn. Fordi etaten har så mange skjema i Altinn har man blitt nødt til å fordele dette arbeidet på flere personer, og ikke kun på skjemaansvarlig. I fagavdelingene hos direktoratet finnes det skjemaansvarlige som har et overordnet ansvar. Under seg har de systemforvaltere som gjør den faktiske jobben med å definere skjema og gjøre den daglige oppfølgingen med skjemaautviklerne.

**IT-avdelingen**

IT-avdelingen har mye ansvar i forhold til Altinn. De holder rede på hva som er mulig i dagens versjon og fremtidige versjoner av Altinn. De rapporterer endringsforslag, ønsker, feil m.m. De har også ansvaret for implementering av skjema i Altinn.

**Teknisk ansvarlig**

Den som er teknisk ansvarlig har en overordnet rolle og jobber på tvers av skjema. Dette er en koordinerende rolle som innebærer samling av krav fra etaten, holde kontakten med leverandør og ASF osv.

**Skjemaautviklere**

Skjemaautviklerne er de som gjør den faktiske jobben med å implementere skjema i Altinn. De tar grunnlaget som er blitt fastsatt av fagavdelingene og lager selve skjemaet i Altinn og legger inn kontroller og feilmeldinger osv.

**Ansvarlig for brukergrensesnitt**

I brukerdokumentasjonen for skjemaautvikling er det også anbefalt å ha en ansvarlig for brukergrensesnitt [Accenture 2004a: 8] (min utheving):

*"I tillegg til disse rollene anbefaler vi å ha **en ansvarlig for brukergrensesnitt** og en **ansvarlig for teknisk utvikling**, som begge opererer på tvers av skjema. Den ansvarlige for brukergrensesnittet bør være en som kan brukergrensesnitt, brukervennlighet og som har øye for detaljer. Vedkommende bør ta en kvalitetssikring*



*av brukergrensesnittet og være ansvarlig for å ta avgjørelser i tilfeller der eksisterende dokumentasjon er mangelfull eller du ønsker å avvike fra standarden.”*

Skattedirektoratet har per i dag ingen som jobber med en koordinerende rolle for brukergrensesnittet i Altinn-skjema.

### **Brukeropplevelsen i Altinn: et todelt ansvar**

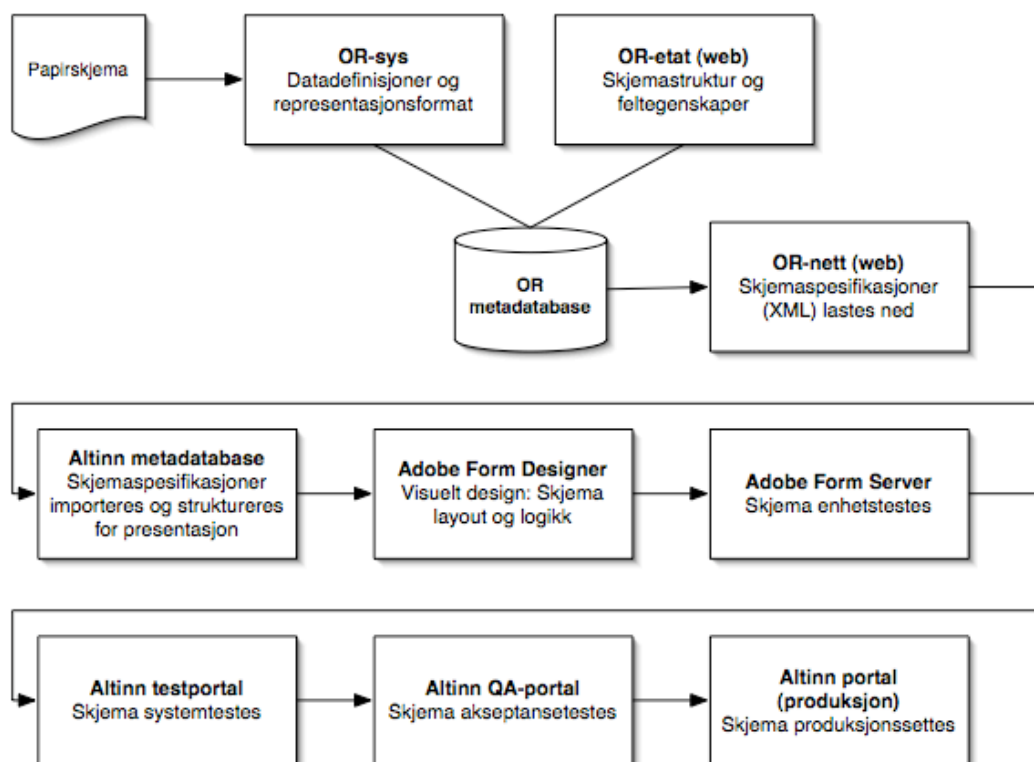
Altinn er fra det offentliges side *todelt* med hensyn på ansvar. Rammeverket til Altinn; selve portalen og hvordan den overordnede løsningen er satt opp og fungerer faller under ansvaret til den koordinerende enheten til Altinn – Altinn sentralforvaltning (ASF).

Mens Altinn sentralforvaltning har i oppgave å ta vare på rammeverket er *det hver enkelt etat* som har ansvar for sine egne skjemaer. Etatene må forholde seg til rammeverket lagt av den tekniske løsningen. Dette betyr at de må forholde seg til retningslinjer og føringer lagt av teknologien, men innenfor disse rammene er det etatene selv som bestemmer utformingen av sine egne skjema.

## 4.5. Prosesser

### Arbeidsprosess for definering av skjema

Definering og oppsett av nettskjema skjer i flere ledd, og i flere verktøy. Først gjøres endringer i ORetat, der felter og ledetekster defineres. Denne første delen av endringene gjøres av skjemaansvarlige og systemforvaltere i fagavdelingene. Når alle endringene er registrert i de forskjellige OR-systemene går jobben over til IT-avdelingen og skjemautviklerne.



**Figur 13: Arbeidsprosess for definering av skjema (gjenspekt fra dokumentasjon for skjemarevisjonen 2004 i SKD)**

Denne oversikten er satt sammen fra flere kilder. Noe er tatt fra "overordnet spesifisering" i Altinn-dokumentasjonen, "Brukerdokumentasjon for skjemaautvikling" fra Accenture og fra dokumentasjon i forbindelse med skjemarevisjonen 2004 i Skattedirektoratet.

#### Definisjon av dataelementer og struktur

Den aktuelle fagavdeling for et gitt skjema jobber mot de forskjellige systemene til Oppgaveregisteret (OR-sys, OR-etat, OR-nett) for å definere skjemaelementer, sette opp struktur og sammenheng for skjemaet m.m. Vi ser av Figur 13 at papirskjema som regel har vært utgangspunkt. For alle eksisterende skjema har papirvarianten vært førende for hvordan det elektroniske skjemaet skulle settes opp. Etter at definisjonene er ferdige lagres de i metadatabasen til Oppgaveregisteret.

##### **Import av data fra OR**

Fra Oppgaveregisterets metadatabase overføres data på Xforms og XML Schema (XSD) format. Dataene versjonsstyres og importeres i Altinns metadatabase.

##### **Visuelt design i Adobe Form Designer**

Dette stadiet kalles for "visuelt design", selv om det involverer både layout og implementasjon av skjemalogikk og navigasjon. Når visuelt design er ferdig lagres endringer i Metadatabasen til Altinn.

Etter at det visuelle designet er ferdig gjøres det flere endringer i Altinns metadatabase. Skjemaet må knyttes til en prosess i Altinn. Prosessene bestemmer hvordan skjemaet skal "oppføre" seg i portalen.

Deretter må det knyttes tilgangskontroll til skjemaet i forbindelse med at ulike roller har ulike rettigheter i forbindelse med utfylling.

Skjema som er definert i OR får preutfyllingsinformasjon for etatsdata fra OR. Denne videreformidles fra metadatabasen til produksjonsmiljøet. Altinn har en lokal kopi av etatsdata i Altinn Register. Altinns metadatabaseeditor er bindeleddet mellom OR og feltene i skjemaet som skal preutfylles.

##### **Test og godkjenning**

Etter at alle stegene over er utført må skjema gjennom en test- og godkjenningsprosess. Skjemadefinisjoner overføres til ulike testmiljøer på ulike filformater under konfigurasjonsstyring. Når skjemaet er godkjent overføres det til produksjonsmiljøet.

## **4.6. Oppsummering**

I dette kapitlet har vi sett at Altinn er ikke bare en komplisert teknisk løsning, det er også en komplisert organisasjon som går på tvers av etater. Vi har nå gått gjennom noen av deltakerorganisasjonene i Altinn og sett på ansvarsområder og arbeidsoppgaver.



## 5. FORBEDRINGSPOTENSIALET I WEBPORTALEN

### 5.1. Introduksjon

Altinn kan sies å være en webapplikasjon mer enn et nettsted i tradisjonell forstand. Hovedformålet for brukerne av Altinn er å fylle ut skjemaer, ikke å finne informasjon ved å klikke seg rundt på nettsider.

I kapittel 4.4 på side 45 så vi at brukeropplevelsen i Altinn er et todelt ansvar. Altinn sentralforvaltning har ansvaret for rammeverket i Altinn, altså webportalen. Etatene har ansvaret for hvordan skjemaene deres er bygget opp. Fordi dette ansvaret er todelt har jeg valgt å dele opp analysen av brukeropplevelsen på samme måte.

Hvis vi går tilbake til definisjonene i kapittel 1.4 ser vi at det er sterk sammenheng mellom begrepene interaksjonsdesign og informasjonsarkitektur. Interaksjonsdesignet kan sies å dekke arbeidsflyten i løsningen; hvordan en bruker finner et skjema, åpner det, fyller det ut, kontrollerer det, og til slutt sender det inn.

Arbeidsflyten i Altinn er bestemt av rammeverket i webportalen. I dette kapittelet vil vi derfor gå gjennom rammeverket og interaksjonsdesignet i applikasjonen. I kapittel 6 vil vi se på etatenes ansvar og informasjonsarkitekturen i skjemaene.

Først presenteres metodene som er blitt brukt, deretter resultatene fra evalueringene som er blitt gjort. Til slutt diskuteres resultatene og løsningsforslag til de identifiserte problemene blir fremmet.

### 5.2. Metode

#### Heuristisk evaluering

Jeg har gjort en heuristisk evaluering [Nielsen og Mack 1994: 25] av Altinn. Heuristisk evaluering er en metode som brukes for å finne problemer i brukergrensesnitt. Metoden brukes gjerne som en del av en iterativ designprosess, der problemene som oppdages blir rettet for så å kjøre nye runder med evaluering av brukergrensesnittet.

Heuristisk evaluering er blant de mest populære evalueringsformene innen fagområdene som berører brukervennlighet, sannsynligvis fordi metoden er enkel, rimelig og kjapp å gjennomføre. Selve metoden går ut på å evaluere et brukergrensesnitt med et lite antall evaluører og et sett med anerkjente retningslinjer for gode grensesnitt (såkalte "heuristikker").

## Gjennomføring

Evalueringen ble gjennomført i mai 2005 på testløsningen av Altinn (<https://altinn1.accenture.no>).

Skjemaet vi har sett på er "RF-1027 – Selvangivelse for selvstendig næringsdrivende". Dette er et av de største og mest komplekse skjemaene som Skatteetaten har.

Jeg har brukt fem personer (heretter kalt evaluatorene) til å evaluere systemet. Nielsen og Mack anbefaler mellom tre og fem evaluatorene siden dette gir det beste utslaget i forhold til kost/nytte [Nielsen og Mack 1994: 33].

Jeg har brukt følgende evaluatorene:

- En profesjonell informasjonsarkitekt/interaksjonsdesigner med over 15 års erfaring med utforming og evalueringer av løsninger med hensyn på brukervennlighet. Har jobbet mye med Altinn og elektroniske skjema.
- En konsulent med fokus på brukervennlighet. Praktisk erfaring fra Altinn som utfyller for eget konsultantselskap og som daglig leder i andre firma. Har også vært interessert i Altinn lenge.
- To masterstudenter i informatikk med spesialisering i brukervennlighet. Den ene har tidligere praktisk erfaring med Altinn som utfyller for eget firma. Den andre har ingen tidligere erfaring med Altinn.
- En bachelorstudent i informatikk, med interesse for brukervennlige løsninger. Ingen tidligere erfaring med Altinn.

Hver evaluator fikk utlevert heuristikkene, og en del praktisk informasjon. De fikk deretter 1-3 timer på seg til å gå gjennom løsningen. Dette ble gjort på egenhånd, slik at funnene til de forskjellige testerne ikke skulle påvirke hverandre.

Etter at evalueringen var gjennomført førte evaluatorene opp funnene i en rapport som så ble sendt til meg. Jeg summerte opp resultatene i en liste og knyttet funnene opp mot spesifikke heuristikker dersom dette ikke allerede var gjort.

Etter dette har jeg gjort en vurdering av alvorlighetsgraden av funnene. Nielsen kaller dette en "severity rating" [Nielsen 19?]. Funnene ble alle vurdert etter Nielsens skala:

- 0: Dette er ikke ansett som et brukervennlighetsproblem
- 1: Kun kosmetisk problem, trenger bare rettes dersom det er tid til overs
- 2: Mindre brukervennlighetsproblem. Retting av dette bør gis lav prioritet
- 3: Stort brukervennlighetsproblem. Retting av dette er viktig og bør få høy prioritet.
- 4: Brukervennlighetskatastrofe. Dette må rettes før lansering av produktet.

Nielsen anbefaler at en slik gradering av feilene gjøres av *alle* evaluatorene (eller i det minste tre stykker), og at man deretter bruker gjennomsnittlig alvorlighetsgrad på hvert enkelt funn. Jeg fikk dessverre ikke gjort dette på grunn av tidshensyn.

Graderingene er dermed kun basert på mitt eget skjønn. Dette betyr at graderingen av feilene blir farget av hvordan *jeg* oppfatter svakheter i Altinn. Ved å bruke gjennomsnittlige graderinger ville min mening ikke hatt like mye å si.

På den annen side trenger ikke dette være negativt. En heuristisk evaluering gjennomføres ofte ved at konsulenter blir leid inn til å gjennomføre selve evalueringen, deretter bedømme alvorlighetsgraden av alle funn. Etter dette er jobben ferdig. Jeg, derimot, har sett på Altinn og problemer med brukeropplevelsen i portalen i nesten ett år. Jeg har jobbet sammen med brukervennlighetseksperter og fagpersonell om problemene i Altinn – både i skjema og i webportalen. Jeg burde derfor ha bygget meg opp en god forståelse for hvilke problemer som er viktige, og dermed ha gode forutsetninger for å gradere resultatene.

### **Heuristikker for god interaksjonsdesign**

Her har jeg samlet en del prinsipper for brukervennlighet. Det er disse prinsippene som har blitt brukt til å gjøre den heuristiske evalueringen av Altinn. Nielsen kaller prinsippene for heuristikker fordi *"These heuristics are general rules that seem to describe common properties of usable interfaces."* [Nielsen og Mack 1994: 28]. Jeg har satt sammen disse retningslinjene basert i hovedsak på tre sett med prinsipper for god utforming av brukergrensesnitt:

- "Ten Usability Heuristics" [Nielsen og Mack 1994: 30]
- "Eight Golden Rules of Interface Design" [Shneiderman og Plaisant 2005: 74]
- "First Principles of Interaction Design" [Tognazzini 2003]

### **1. Konsistens**

Konsistens er viktig for at brukeren skal slippe å måtte sette seg inn i nye situasjoner i systemet. Gjennom å lage konsistente systemer fremstår de helhetlige, og brukeren kan bruke tidligere kunnskap til å løse nye problemer.

Unødig inkonsistens er også viktig for ikke å forvirre brukeren. Inkonsistens som ikke gir mening gjør at brukeren stopper og bruker kognitiv kapasitet på noe som ikke fungerer som forventet i forhold til tidligere erfaring.

[Shneiderman]:

#### ***Strive for consistency***

*Consistent sequences of actions should be required in similar situations; identical terminology should be used in prompts, menus, and help screens; and consistent commands should be employed throughout.*

[Tognazzini]:

*It is just as important to be visually inconsistent when things must act differently as it is to be visually consistent when things act the same.*

[Nielsen]:

#### ***Consistency and standards***

*Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.*

## **2. Gjør det enkelt å finne det du trenger**

Før du i det hele tatt kan bruke noe, må du *finne* det. På nettet som helhet må brukeren først finne det aktuelle nettstedet. På nettstedets- / applikasjonsnivå er det snakk om å gjøre det enkelt for brukeren å finne det han trenger å gjøre. De vanligste virkemidlene for å finne fram er navigasjon og søk.

[Morville 2004a]:

### ***Design for findability***

*Findability refers to the quality of being locatable or navigable. At the item level, we can evaluate to what degree a particular object is easy to discover or locate. At the system level, we can analyze how well a physical or digital environment supports navigation and retrieval.*

## **3. Lag gode dialoger med klar avslutning**

[Shneiderman]:

### ***Design dialog to yield closure***

*Sequences of action should be organized into groups with a beginning, middle, and end. The informative feedback at the completion of a group of actions gives the operators the satisfaction of accomplishment, [...] and an indication that the way is clear to prepare for the next group of actions.*

## **4. La brukeren være i kontroll**

Systemet bør støtte *brukerens* mål og oppgaver, og burde følgelig la brukeren ta styringen. Det er altså brukeren som bør diktere flyten i systemet, og ikke omvendt.

[Shneiderman]:

### ***Support internal locus of control***

*Experienced operators strongly desire the sense that they are in charge of the system and that the system responds to their actions. Design the system to make users the initiators of actions rather than the responders.*

## **5. Tilgivende systemer og angremuligheter**

Systemer som tar høyde for, og tillater, at brukeren gjør feil oppleves som mer behagelige å bruke. Dette er viktig for at brukerne skal føle seg avslappet i bruken av systemet.

[Tognazzini]:

### ***Explorable interfaces***

- *Make actions reversible.*
- *Always allow "undo".*



- *Always allow a way out.*

[Shneiderman]:

***Permit easy reversal of actions***

*This feature relieves anxiety, since the user knows that errors can be undone; it thus encourages exploration of unfamiliar options. The units of reversibility may be a single action, a data entry, or a complete group of actions.*

[Nielsen]:

***User control and freedom***

*Users often choose system functions by mistake and will need a clearly marked 'emergency exit' to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.*

## **6. Redusér byrden på brukeren**

Der det er mulig burde *systemet* og ikke brukeren bære byrden. Brukeren bør for eksempel ikke måtte gjøre beregninger, eller huske tall fra en del av systemet til et annet. Det er systemet som skal hjelpe brukeren og ikke omvendt.

[Shneiderman]:

***Reduce short-term memory load***

*The limitation of human information processing in short-term memory requires that displays be kept simple, multiple page displays be consolidated, window-motion frequency be reduced, and sufficient training time be allotted for codes, mnemonics, and sequences of actions.*

[Nielsen]:

***Recognition rather than recall***

*Make objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.*

## **7. Sysselsett brukeren**

Ved å sysselsette brukeren føler han at han *gjør* noe. Denne følelsen av å gjøre noe gjør at brukeren føler en nytteverdi ved å bruke systemet. Sysselsett derfor brukeren i størst mulig grad, og la brukeren vente minst mulig – la heller systemet gjøre prosesseringer i bakgrunnen.

[Tognazzini]:

***Keep the user occupied***

[Tognazzini]:

***Latency reduction***

*Latency can often be hidden from users through multitasking techniques, letting them continue with their work while transmission and computation take place in the background.*

## **8. Ta vare på brukerens arbeid**

Ved bruk av datamaskiner er det få ting som er mer frustrerende enn å miste arbeid. Systemene vi bruker bør derfor gå langt i å ta vare på det vi plotter inn i dem.

[Tognazzini]:

### ***Protect users' work***

*Ensure that users never lose their work as a result of error on their part, the vagaries of internet transmission, or any other reason other than the completely unavoidable, such as sudden loss of power to the client computer.*

[Tognazzini]:

### ***Track state***

*State information should be held in a cookie on the client machine during a session with a transaction service, then stored on the server when they log off.*

*Users should be able to log off at work, go home and take up exactly where they left off.*

## **9. Status og tilbakemeldinger**

Status og tilbakemeldinger er kritisk for at brukeren skal skjønne hva som foregår og at systemet svarer på hans kommandoer.

[Shneiderman]:

### ***Offer informative feedback***

*For every operator action, there should be some system feedback. For frequent and minor actions, the response can be modest, while for infrequent and major actions, the response should be more substantial.*

[Nielsen]:

### ***Visibility of system status***

*The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.*

[Tognazzini]:

### ***Use status mechanisms to keep users aware and informed***

*[...] Status mechanisms are vital to supplying the information necessary for workers to respond appropriately to changing conditions. [...] Users should not have to seek out status information. Rather, they should be able to glance at their work environment and be able to gather at least a first approximation of state and workload.*

## 10. Håndtér feil godt

Uansett hvor godt et system er designet vil brukeren gjøre feil. Når feilen først skjer er det viktig å håndtere disse situasjonene på en best mulig måte slik at brukeren kan komme seg ut av situasjonen og gå videre med arbeidet sitt.

[Shneiderman]:

### ***Offer simple error handling***

*As much as possible design the system so the user cannot make a serious error. If an error is made, the system should be able to detect the error and offer simple, comprehensible mechanisms for handling the error.*

[Nielsen]:

### ***Help users recognize, diagnose, and recover from errors***

*Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.*

## 11. Avverging av feil

[Nielsen]:

### ***Error prevention***

*Even better than good error messages is careful design which prevents a problem from occurring in the first place.*

## 12. Hjelp og dokumentasjon

[Nielsen]:

### ***Help and documentation***

*Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.*

## 13. Snarveier

[Shneiderman]:

### ***Enable frequent users to use shortcuts***

*As the frequency of use increases, so do the users desires to reduce the number of interactions and to increase the pace of interaction. Abbreviations, function keys, hidden commands, and macro facilities are very helpful to an expert user.*

[Nielsen]:

***Flexibility and efficiency of use***

*Accelerators – unseen by the novice user – may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.*

**14. Fitt's lov**

[Tognazzini]:

*The time to acquire a target is a function of the distance to and size of the target.*

Denne loven sier at jo større et brukergrensesnittelement er, jo lettere er det å treffe den. Knapper burde for eksempel ha fornuftig avstand og være store nok til at de er lette å treffe. Dette gjør systemet lettere å bruke for alle – men spesielt for de som har problemer med å bruke mus.

## 5.3. Resultater

### Heuristisk evaluering

Under følger funn fra den heuristiske evalueringen. De står listet opp i samme rekkefølge som kriteriene vi har brukt. Mange av funnene passer under flere kriterier og er derfor listet opp flere ganger. Tallet i parentes bak feilen representerer antall evaluatorene som har funnet feilen. Det er funnet feil under alle de fjorten kategoriene. Det er også kommet til to nye kategorier som evaluatorene har lagt til under evalueringen.

#### 1. Konsistens

- Ved innlogging er det problematisk å skille mellom valgene av engangskoder. Hva er forskjellen på engangskoder i SMS og engangskoder i brev? (1)
- Spørsmål om lagring ved navigering fra vedlegg- til hovedskjema har motsatt logikk i forhold til samme situasjon for navigering fra hovedskjema til hovedside. (2)
- Det er lite konsistens i hva som kalkuleres og hva som ikke kalkuleres. (1)
- Hjelpeikonet er klikkbart over alt inne i skjemaet. Men i resten av Altinn er det ikke klikkbart (man må klikke på teksten ved siden av). (1)
- Når man har foretatt skjemakontroll og fortsetter arbeidet vil feilflippen vises. Dersom man velger å vise hjelpeflippen på en side, for så å gå til neste side vil feilflippen komme opp nok en gang. (1)
- I arkivet er feilmeldingen annerledes enn feilmeldingene man får inne i skjemaene. Her får man inline-feilmelding (ikke dialogboks). Det er heller ikke noen automatisk konvertering av tallene man taster inn. (1)

#### 2. Gjør det enkelt å finne det brukerne trenger

- Man kan finne skjemaet man leter etter i portalen (før man logger inn). Men du kan ikke gå rett til utfylling fra portalen. Du må logge inn, og finne skjemaet på nytt etter at du har logget inn. (1)
- Man kan kun søke fra skjemakatalogen, ikke fra hovedsiden. (1)
- Man må søke på svært spesifikke ord for å finne skjema. Man kan ikke skrive for mange ord, selv om de finnes i skjemanavnet. Søk på "Selvangivelse for næringsdrivende 2004" gav f.eks ingen treff, mens søk på "selvangivelse for næringsdrivende" gav treff. (1)
- Søk på tema eller innhold i skjema mangler. (1)
- Vedlegg og ekstra tabeller er ikke lett å finne (før du har lært det). (2)
- Man må vite hvilket skjema man skal fylle ut. Og man må vite hvilken etat som har det. (2)
- Selv om man vet hvilket skjema man skal fylle ut er det vanskelig å finne det dersom det er et underskjema til et overordnet skjema som selvangivelsen. Å søke på "næringsoppgave" fungerer f.eks ikke. (2)
- Feil i summeringsfelt ved skjemakontroll sier ingenting om hvor feilen egentlig stammer fra. Det er opp til brukeren selv å finne fram til feltet som egentlig har skyld i feilen. (1)

- Det står flere steder at man kan se i rettleddningen dersom man har spørsmål. Men å finne rettleddningen er ikke enkelt. (2)

### 3. Lag gode dialoger med klar avslutning

- Det er uklart hva som er avslutning av skjemautfylling og hva som er overgang til innsending. (1)
- Ved bruk av vedleggstabeller føles det som om man blir kastet ut av skjemaet. ("Hjelp! Jeg er blitt kastet ut! Hvor er angre-knappen?!") (1)
- På hovedsiden er det ikke umiddelbart klart hvor man skal gå for å "opprette" nye skjemaer. (1)
- Ved innlogging er det lett å overse de alternative valgene for innlogging på høyresiden (valg B og C). Fokuset går mot forms-elementene. (1)
- Portalen (før innlogging) inneholder for mye informasjon. (1)
- I feillisten etter skjemakontroll står utdypende forklaring til feilen i mouseover-tekst. Denne vises ikke lenge nok til at man får lest feilen. Følgelig må man flytte pekeren fram og tilbake flere ganger for å få med seg feilen. (1)
- Ved skjemakontroll får man beskjed om at skjema for beregning av personinntekt ikke er opprettet. Hvorfor henvises man ikke til der man kan opprette skjemaet, eller får spørsmål om man vil opprette skjemaet med en gang? (1)
- Når man legger til instanser i en vedleggstabell (f.eks arbeidsgivere) og går tilbake til selve skjemaet er det vanskelig å se at tallene der har blitt oppdatert. Kun ved utregning av sum-feltet i den aktuelle posten vet man at dataene har blitt lagret. De nye instansene burde ligge rett i skjemaet. Det gir bedre oversikt. (1)
- Når man flytter et skjema til hovedsiden får man ikke god tilbakemelding på at noe har skjedd. Man blir stående på samme siden. (1)
- Hvorfor må skjema "flyttes til hovedside"? Kan ikke brukeren flyttes til hovedsiden eller skjemaet direkte? Brukeren vil bare fylle ut skjemaet. (1)
- Dialogbokser har inkonsistente valg ved lagring/avslutning av hovedskjema og vedleggsskjema. (1)
- Når man retter feil og har feilflippen åpen går det ikke an å klikke på hjelpeikoner. Det kommer da opp en dialogboks som ber deg klikke frem hjelpeflippen manuelt. (3)

### 4. La brukeren ha kontrollen

- Selv om man vet hvilket skjema man skal fylle ut er det vanskelig å finne det dersom det er et underskjema til et overordnet skjema som selvangivelsen. Å søke på "næringsoppgave" fungerer f.eks ikke. Brukeren må legge til det overordnede skjemaet, og deretter (via hovedskjemaet) legge til de aktuelle vedleggsskjema. (2)
- Brukeren må vite et spesifikt navn eller hvilket etat som har skjemaet. Innholdet er avsenderorientert og ikke brukerorientert. (2)
- Etter å ha gjennomført kontroll av skjema kommer man tilbake til første siden i skjemaet. (2)

- Når man retter feil og har feilflippen åpen går det ikke an å klikke på hjelpeikoner. Det kommer da opp en dialogboks som ber deg klikke frem hjelpeflippen manuelt. (3)
- Når man har foretatt skjema kontroll og fortsetter arbeidet vil feilflippen vises. Dersom man velger å vise hjelpeflippen på en side, for så å gå til neste side vil feilflippen komme opp nok en gang. (1)
- Når man skal legge til vedleggsskjema kommer det opp en nedtrekksliste med ubrukelige navn (hvis man ikke kjenner RF-kodene til skjemaene). Man må trykke på "utvid" for å få brukbare nedtrekkslister i Internet Explorer (Safari og Firefox har bedre nedtrekkslister som er tilpasset lengden på navnene). (1)
- Man står fortsatt i skjema-bildet etter automatisk utlogging. Man kan altså potensielt fortsette å fylle ut resten av siden, for så å trykke "neste". Først da får man beskjed om at man må logge inn, og det man har gjort på siden blir borte. (1)
- Hovedsiden burde gi brukeren valg ut i fra tildelte roller (f.eks. Rapportering for deg selv som privatperson, eget firma, daglig leder i annet firma osv) (1)
- Det er vanskelig å finne kontaktinformasjon til faglig hjelp for utfylling av skjema. (1)

#### 5. Tilgivende systemer og angremuligheter

- Ingen angremuligheter ved utfylling. (1)
- Ved bruk av vedleggstabeller føles det som om man blir kastet ut av skjemaet. ("Hjelp! Jeg er blitt kastet ut! Hvor er angre-knappen?!") (Fant ikke "tilbake til skjema"-knappen. (1)
- Å lete seg fram til et skjema ved å trykke på "vis" er tregt og ustabilt. PDFene som kommer opp vises ofte ikke og Windows kræsjer. (1)
- Dialogbokser har inkonsistente valg ved lagring/ avslutning av hovedskjema og vedleggsskjema. (1)

#### 6. Redusér byrden på brukeren

- Innloggingen foregår over flere trinn og er omstendelig. (1)
- Hvorfor er engangskodene følsomme ovenfor store og små bokstaver? Det er kronglete å taste inn, og lett å taste feil. (1)
- Portalen (før innlogging) inneholder for mye informasjon. (1)
- Ved skjema kontroll får man beskjed om at skjema for beregning av personinntekt ikke er opprettet. Hvorfor henvises man ikke til der man kan opprette skjemaet, eller får spørsmål om man vil opprette skjemaet med en gang? (1)
- Når man legger til instanser i en vedleggstabell (f.eks arbeidsgivere) og går tilbake til selve skjemaet er det vanskelig å se at tallene der har blitt oppdatert. Kun ved utregning av sum-feltet i den aktuelle posten vet man at dataene har blitt lagret. De nye instansene burde ligge rett i skjemaet. Det gir bedre oversikt. (1)
- Skjema utfører en del beregninger, men tall må flyttes manuelt mellom skjema (og av og til internt i skjema). (1)

- Brukeren må vite et spesifikt navn eller hvilket etat som har skjemaet. Innholdet er avsenderorientert og ikke brukerorientert. (2)
- Etter å ha gjennomført kontroll av skjema kommer man tilbake til første siden i skjemaet. (1)
- Feil i summeringsfelt ved skjemakontroll sier ingenting om hvor feilen egentlig stammer fra. (1)
- Når man har valgt en hjelpetekst i et skjema er det vanskelig å se hvilken tekst som er valgt. Man har ingen indikasjon på hvilket punkt man leser på annet enn en forkortet overskrift. (1)
- Etter en viss tid uten aktivitet logges man ut. Når man klikker på en link vil man da bli bedt om å logge inn på nytt. Etter innlogging kommer man til første side i skjemaet, og ikke til den siden man faktisk var på. (1)
- Det står flere steder at man kan se i rettledningen dersom man har spørsmål. Men å finne rettledningen er ikke enkelt. (2)
- Det er vanskelig å finne kontaktinformasjon til faglig hjelp for utfylling av skjema. (1)
- Det er for mange liknende valg når man søker etter hjelp: "Hjelp", "Hjelpetekst", "Kontakt Altinn" og "Om skjema" Det er ikke opplagt for brukeren hvor han henvender seg for hva. (1)
- Vanskelig å huske hvor man skal gå for å bytte rolle. (1)
- Venstremenyen består av korte overskrifter som kuttet. En del av plassen brukes også av tallene. Brukeren får kun hele overskriften ved å holde musepekeren over overskriftene. (1)

## 7. Sysselsett brukeren

- På sider med mange felter tar det av og til en del tid før neste side kommer opp ved klikk på forrige/neste-knapper. Denne ventingen gjør at man tidvis kvier seg for å gå frem og tilbake, selv om det med dagens oppbygning av skjema er nødvendig. (1)

## 8. Ta vare på brukerens arbeid

- Lagreknapp er sentralt plassert, men det er uklart hvor ofte den må brukes eller om andre funksjoner lagrer i bakgrunnen. (3)
- Man blir bedt om å lagre eksplisitt når man går ut av et skjema. Hvis man ikke lagrer eller klikker seg videre blir det man har fylt ut ikke lagret. (1)
- Etter en viss tid uten aktivitet logges man ut. Når man klikker på en link vil man da bli bedt om å logge inn på nytt. Etter innlogging kommer man til første side i skjemaet, og ikke til den siden man faktisk var på. (1)
- Man står fortsatt i skjema-bildet etter automatisk utlogging. Man kan altså potensielt fortsette å fylle ut resten av siden, for så å trykke "neste". Først da får man beskjed om at man må logge inn, og det man har gjort på siden blir borte. (1)

## 9. Status og tilbakemeldinger

- Det finnes ingen indikasjoner på hvilke sider som er utfylt/ferdig i utfyllingssituasjonen. (1)



- Ved bruk av vedleggstabeller føles det som om man blir kastet ut av skjemaet. ("Hjelp! Jeg er blitt kastet ut! Hvor er angre-knappen?!") (1)
- Det er vanskelig å vite om det er lenge igjen til timeout mot serveren. Dette er stressende. (1)
- Hvorfor puttes viktig informasjon (hjelp, kontakt osv) øverst i høyre hjørne der det er vanskelig å få øye på den? (1)
- Etter å ha brukt veiviseren for innføring i Altinn var det rart å ikke se at det var noen fremdriftsindikator i selve skjemaet. (Skjønste ikke før et par sider ut i skjemaet at venstremenyen fungerer som en fremdriftsindikator). (1)
- Når man legger til instanser i en vedleggstabell (f.eks arbeidsgivere) og går tilbake til selve skjemaet er det vanskelig å se at tallene der har blitt oppdatert. Kun ved utregning av sum-feltet i den aktuelle posten vet man at dataene har blitt lagret. De nye instansene burde ligge rett i skjemaet. Det gir bedre oversikt. (2)
- Når man fyller ut skjemaet i tilfeldig rekkefølge er det av og til vanskelig å skjønne at enkelte felter henter data fra tidligere felter i skjemaet. (1)
- Disabled-felter er vanskelig å skille fra vanlige felter. Dette gjelder spesielt på skjermer med lav kontrast (som f.eks laptop-skjermer). (3)
- Når man har valgt en hjelpetekst i et skjema er det vanskelig å se hvilken tekst som er valgt. Man har ingen indikasjon på hvilket punkt man leser på annet enn en forkortet overskrift. (1)
- Når man skal legge til vedleggsskjema kommer det opp en nedtrekksliste med ubrukelige navn (hvis man ikke kjenner RF-kodene til skjemaene). man må trykke på "utvid" for å få brukbare nedtrekkslister i Internet Explorer (Safari og Firefox har bedre nedtrekkslister som er tilpasset lengden på navnene). (1)
- Man står fortsatt i skjema-bildet etter automatisk utlogging. Man kan altså potensielt fortsette å fylle ut resten av siden, for så å trykke "neste". Først da får man beskjed om at man må logge inn, og det man har gjort på siden blir borte. (1)
- Ved klikk på "kontakt Altinn" åpnes et nytt vindu av Altinn-informasjonsportalen som legger seg over skjemaet. I dette vinduet vises logg inn-knappene slik at det ser ut som om brukeren er logget ut. (1)
- Hvem/hva man rapporterer for burde komme fram tydeligere. (1)
- Ved signering er navnet på skjemaet som skal signeres skrevet i "utgrået" tekst. Dette er feil bruk av virkemiddel siden "utgrået" tekst skal brukes ved ikke-relevant informasjon. (1)
- Tilbakemeldinger på utført oppgave mangler, eller er for lite synlig. For eksempel: flytte skjema til hovedsiden fra skjemakatalog. (2)
- Venstremenyen består av korte overskrifter som kuttes. En del av plassen brukes også av tallene. Brukeren får kun hele overskriften ved å holde musepekeren over overskriftene. Dette gir dårlig oversikt over innholdet i skjemaet. (1)

### 10. Håndtér feil godt

- Mange feil gis det kun beskjed om ved innsending/kontroller skjema. Dette gir liten sammenheng med når feilen oppstod. (2)

- Feil i overførte eller kalkulerte felt (f.eks om manglende utfylling) angis i felt som ikke er skrivbare. Det er heller ikke mulig å vite hvor feilen stammer fra. (3)
- Umiddelbare feilmeldinger (og øvrige tilbakemeldinger) vises som dialogbokser. Disse ser ut som generelle systemfeil fra Pcen, ikke som en melding fra Altinn. (1)
- I feillisten etter skjemakontroll kommer detaljer om feilen opp som mouseover-tekst. Denne teksten blir i enkelte nettlesere forkortet slik at man ikke får med meningsinnholdet. Dette skjer f.eks i Firefox (Mac). (1)
- I feillisten etter skjemakontroll står utdypende forklaring til feilen i mouseover-tekst. Denne vises ikke lenge nok til at man får lest feilen. Følgelig må man flytte pekeren fram og tilbake flere ganger for å få med seg feilen. (1)
- Det går an å legge til flere identiske skjema i skjemalisten. Finnes det tilfeller der man trenger det? (1)

### **11. Avverging av feil.**

- Ved innlogging er det lett å overse de alternative valgene for innlogging på høyresiden (valg B og C). Blikket trekkes mot forms-elementene. (1)
- Veiviseren i portalen kom ikke opp i det hele tatt i Firefox (både Windows og Mac) og Safari (Mac). (2)
- Man får ikke mye hjelp til å finne ut hvilke(t) skjema(er) man skal fylle ut. Brukere kan dermed potensielt fylle ut feil skjema. Det finnes mange like skjema /skjemanavn. (1)
- Det er ikke mulig å legge inn dato i alternative format (dette varierer litt). (1)
- Årstall med to siffer tolkes feil, f.eks tolkes 85 til 2085 istedenfor 1985. (2)
- Det mangler hint om inntastingsformat flere steder. (2)

### **12. Hjelp og dokumentasjon**

- Veiviseren i portalen kom ikke opp i det hele tatt i Firefox (både Windows og Mac) og Safari (Mac). (2)
- Man får ikke mye hjelp til å finne ut hvilke(t) skjema(er) man skal fylle ut. (1)
- Feil i summeringsfelt ved skjemakontroll sier ingenting om hvor feilen egentlig stammer fra. Man får heller ingen hjelp til feilsøk når man klikker på hjelpeikonet til det aktuelle feltet. (1)
- Mye av hjelpteksten gir for lite hjelp i forhold til hva brukeren lurte på i utfyllingsøyeblikket. (2)
- Det står flere steder at man kan se i rettledningen dersom man har spørsmål. Men å finne rettledningen er ikke enkelt. (1)
- Rettledningen er i PDF-format (1.6 MB) og ikke spesielt enkel å bruke for raske oppslag på skjerm. (1)

### **13. Snarveier**

- Det finnes svært få (ingen?) snarveier i løsningen. (1)
- Ved skjemakontroll av selvangivelsen kan man få beskjed om at skjema for beregning av personinntekt ikke er opprettet. Hvorfor henvises man ikke til der man kan opprette skjemaet, eller får spørsmål om man vil opprette skjemaet med en gang? (1)

- Hvorfor må skjema "flyttes til hovedside"? Kan ikke brukeren flyttes til hovedsiden eller skjemaet direkte? Brukeren vil bare fylle ut skjemaet. (1)

#### **14. Fitts lov**

- Generelt er det liten visuell indikasjon på hva som er det viktigste på en side. Sider som viser alternativer er uklart utformet. (1)
- Tilhørende tekst til radioknapper er ikke klikkbare. (1)
- Det er lett å bomme i venstremenyen og klikke på et annet valg enn det som var tiltenkt. (1)

#### **15. Tilpassning (ny)**

- Hvorfor må alle brukere innom "barns lønnsinntekt" og "mottatte livrenter"? Brukere må ta stilling til ting som mottaker enten vet fra før (og da kan stille spørsmål om) eller som man kjapt kunne svart ja/nei på i begynnelsen av skjemaet. (1)

#### **16. Affordances (ny)**

- Venstremenyen i skjemaene gir ikke gode nok signaler om at de er klikkbare. (1)
- Venstremenyen er navigerbar, men gir ikke gode nok signaler om data blir lagret. (Er vant til statiske menyer på nettsteder der forms IKKE blir lagret ved klikk på meny). (1)

## 5.4. Diskusjon

### Heuristisk evaluering

Jeg har gjort en gradering av alvorligheten på funnene. Resultatene som diskuteres her har alvorlighetsgrad 3 (stort brukervennlighetsproblem) eller 4 (brukervennlighetskatastrofe). Andre feil har jeg utelatt, siden disse ikke anses som kritiske nok.

Funnene fra den heuristiske evalueringen er delt i to. *Funn som gjelder webportalen og rammeverket diskuteres her.* Funn som gjelder innholdet i skjemaet diskuteres i kapittel 6, som ser på forbedringspotensialet i skjemaformet.

Jeg har valgt å sortere funnene inn i grupper, slik at funn som berører samme tema holdes samlet. Jeg starter med en oppsummert liste av funnene først, for deretter å diskutere de samlet innen hver gruppe. De feilene som anses som mest kritiske i hver gruppe blir som regel diskutert først. I enkelte tilfeller har jeg diskutert et mindre kritisk punkt først der dette har vært nødvendig for å gi en innføring i problematikken rundt funnene.

### Oppsummerte funn

#### Findability

1. Brukeren må vite et spesifikt navn eller hvilken etat som har skjemaet. Innholdet er avsenderorientert og ikke brukerorientert.
  - Bryter med: 6) *brukerbyrde*, 4) *brukerkontroll*
  - Alvorlighetsgrad: 3
2. Søk på tema eller innhold i skjema mangler.
  - Bryter med: 2) *finnbarhet*
  - Alvorlighetsgrad: 3
3. Man må søke på svært spesifikke ord for å finne skjema. Man kan ikke skrive for mange ord, selv om de finnes i skjemanavnet. Søk på f.eks. "selvangivelse for næringsdrivende 2004" gav ingen treff, mens søk på "selvangivelse for næringsdrivende" gav treff.
  - Bryter med: 2) *finnbarhet*
  - Alvorlighetsgrad: 3

#### Navigasjon og oversikt

4. Venstremenyen består av korte overskrifter som kuttet. En del av plassen brukes også av tallene. Brukeren får kun hele overskriften ved å holde musepekeren over overskriftene. Dette gir dårlig oversikt over innholdet i skjemaet.
  - Bryter med: 6) *brukerbyrde*, 9) *status*
  - Alvorlighetsgrad: 4

#### Håndtering av vedlegg

5. Selv om man vet hvilket skjema man skal fylle ut er det vanskelig å finne det dersom det er et underskjema til et overordnet skjema som selvangivelsen. Å

søke på "næringsoppgave" fungerer f.eks. ikke. Brukeren må legge til det overordnede skjemaet, og deretter (via hovedskjemaet) legge til de aktuelle vedleggsskjema.

- Bryter med: 4) *brukerkontroll*, 2) *finnbarhet*
  - Alvorlighetsgrad: 4
6. Vedlegg og ekstra tabeller er ikke lett å finne (før du har lært det).
- Bryter med: 2) *finnbarhet*, 1) *konsistens*
  - Alvorlighetsgrad: 3
7. Når man legger til instanser i en vedleggstabell (f.eks. arbeidsgivere) og går tilbake til selve skjemaet er det vanskelig å se at tallene der har blitt oppdatert. Kun ved utregning av sum-feltet i den aktuelle posten vet man at dataene har blitt lagret og overført til hovedskjemaet. Dette gjør brukeren usikker.
- Bryter med: 6) *brukerbyrde*, 3) *gode dialoger*, 9) *status*
  - Alvorlighetsgrad: 3

### Lagring av brukers arbeid

8. Spørsmål om lagring ved navigering fra vedlegg- til hovedskjema har motsatt logikk i forhold til samme situasjon for navigering fra hovedskjema til hovedside.
- Bryter med: 1) *konsistens*
  - Alvorlighetsgrad: 3
9. Man blir eksplisitt bedt om å lagre når man går ut av skjema. Hvis man ikke lagrer eller klikker seg videre blir det man har fylt ut ikke lagret.
- Bryter med: 8) *ivareta arbeid*
  - Alvorlighetsgrad: 3
10. Man står fortsatt i skjema-bildet etter automatisk utlogging. Man kan altså potensielt fortsette fylle ut resten av siden, for så å trykke på "neste". Først *da* får man beskjed om at man må logge inn, og det man har gjort på foregående side blir borte.
- Bryter med: 4) *brukerkontroll*, 8) *ivareta arbeid*, 9) *status*
  - Alvorlighetsgrad: 4
11. Venstremenyen i skjemaene gir ikke gode nok signaler om at den er klikkbar.
- Bryter med: *affordances (ny)*
  - Alvorlighetsgrad: 3
12. Lagreknapp er sentralt plassert, men det er uklart hvor ofte den må brukes eller om andre funksjoner lagrer i bakgrunnen.
- Bryter med: 8) *ivareta arbeid*
  - Alvorlighetsgrad: 3

### Hjelpesystemet

13. Det er mange liknende valg når man søker etter hjelp: "hjelp", "hjelpetekst", "kontakt Altinn" og "Om skjema". Det er ikke opplagt for brukeren hvor han skal henvende seg for hva.
- Bryter med: 6) *brukerbyrde*
  - Alvorlighetsgrad: 3
14. Det står flere steder at man kan se i rettleidingen dersom man har spørsmål. Men det å finne rettleidingen er ikke enkelt.
- Bryter med: 6) *brukerbyrde*, 2) *finnbarhet*, 12) *hjelp*
  - Alvorlighetsgrad: 4
15. Det er vanskelig å finne kontaktinformasjon til faglig hjelp for utfylling av skjema.
- Bryter med: 6) *brukerbyrde*, 4) *brukerkontroll*
  - Alvorlighetsgrad: 3

16. Veiviseren i portalen kom ikke opp i det hele tatt i Firefox (både Mac og Windows) og Safari (Mac).
  - Bryter med: 11) *feilavverging*, 12) *hjelp*
  - Alvorlighetsgrad: 3

### Feilretting

17. Når man retter en feil og har feilflippen åpen går det ikke an å klikke på hjelpeikoner. Det kommer da opp en dialogboks som ber deg klikke frem hjelpeflippen manuelt.
  - Bryter med: 4) *brukerkontroll*, 3) *gode dialoger*
  - Alvorlighetsgrad: 3
18. I feillisten etter skjemakontroll står utdypende forklaring til feilene i mouseover-tekst. Denne vises ikke lenge nok til at man får lest hele feilmeldingen. Følgelig må man flytte pekeren frem og tilbake flere ganger for å få med seg feilen. Den er heller ikke synlig før man holder musepeker rolig over linken, alle vil derfor ikke finne den.
  - Bryter med: 10) *feilhåndtering*, 3) *gode dialoger*, 9) *status*
  - Alvorlighetsgrad: 3
19. I feillisten etter skjemakontroll kommer detaljer om feilen opp som mouseover-tekst. Denne teksten blir i enkelte nettlesere forkortet, slik at man ikke får med seg meningsinnholdet. Dette skjer f.eks. i Firefox (Mac).
  - Bryter med: 10) *feilhåndtering*
  - Alvorlighetsgrad: 3
20. Disabled-felter (ikke-skrivbare felter) er vanskelig å skille fra vanlige felter. Dette gjelder spesielt på skjermer med lavere kontrast (som f.eks. skjermer på bærbare).
  - Bryter med: 9) *status*
  - Alvorlighetsgrad: 3

## Analyse og løsningsforslag

Her vil jeg diskutere de forskjellige funnene fra den heuristiske evalueringen som jeg oppsummerte over. Jeg vil også komme inn på ytterlige forbedringer av Altinn-rammeverket her. Disse ytterligere forbedringene er flettet inn under de forskjellige funnene der det har vært naturlig å diskutere forbedringer. Forbedringer innen findability kommer følgerig under findability-punktet osv.

### Findability

#### Avsenderorientert skjemaorganisering

1. Brukeren må vite et spesifikt navn eller hvilken etat som har skjemaet. Innholdet er avsenderorientert og ikke brukerorientert.
  - Bryter med: 6) *brukerbyrde*, 4) *brukerkontroll*
  - Alvorlighetsgrad: 3

Organiseringen av skjemakatalogen og søke-funksjonen i skjemakatalogen tar utgangspunkt i en organisering etter etater og skjemanummer (såkalte RF-nummer). Dette gjelder også i lister – vedleggslisters som kommer opp som nedtrekkslisters er sortert etter RF-nummer. Dette er en såkalt "eksakt organiseringmåte" [Rosenfeld og Morville 2002: 56]

Å si at organiseringen av skjemaene ikke er brukerorientert er kanskje litt for sterkt. For de brukerne som jobber *profesjonelt* med Skatteetatens skjema (f.eks. regnskapsførere og revisorer) er nok dette en foretrukket måte å organisere skjema på. Disse menneskene kan ofte RF-nummerne på rams, og i de tilfellene de vet hvilket skjema de trenger fungerer dette godt. Dette kalles "known-item searching" [ibid: 56].

For *sporadiske brukere*, derimot; privatpersoner og de som driver små bedrifter som enkeltpersonsforetak o.l., er en slik organisering ikke like lett å bruke. Denne brukergruppen kan godt vite hvilket skjema de trenger å fylle ut, men de trenger ikke nødvendigvis huske RF-nummeret til skjemaet. For disse brukerne burde det være mulig å finne skjemaer på andre måter.

Den mest logiske måten å sortere skjema på må være en alfabetisk oversikt, som da starter med *navnet* på skjemaet og ikke RF-nummer slik opplister vises i dag. Det kunne også være greit å få se alle skjema uavhengig av hvilket etat som eier det. Det kan være tilfeller der brukeren ikke er sikker på hvilken etat som har et skjema. Dette er mer vanlig i tilfeller der brukerne ikke er helt sikre på hva de er ute etter. Dette kalles utforskende leting ("exploratory seeking" [ibid: 32]).

For å ytterlige støtte utforskende leting kan skjema også sorteres etter "ambiguous organization schemes" [ibid: 56]. Eksempler på slik organisering kan være etter *emne*,

*publikum* eller *oppgave*. For eksempel kan det settes opp emner for selvangivelse, og selvangivelse for næringsdrivende.

For at slike opplister skal bli mer nyttige burde også flere skjema listes opp det gjøres i dag. I dagens skjemakatalog listes kun hovedskjema opp, ingen vedleggsskjema. Jeg kommer tilbake til dette senere.

### Søke etter skjema

Jeg behandler to funn samlet her, siden de henger sammen.

#### 2. Søk på tema eller innhold i skjema mangler.

- Bryter med: 2) *finnbarhet*
- Alvorlighetsgrad: 3

3. Man må søke på svært spesifikke ord for å finne skjema. Man kan ikke skrive for mange ord, selv om de finnes i skjemanavnet. Søk på f.eks. "selvangivelse for næringsdrivende 2004" gav ingen treff, mens søk på "selvangivelse for næringsdrivende" gav treff.

- Bryter med: 2) *finnbarhet*
- Alvorlighetsgrad: 3

I tillegg til å lage selve katalogen tilgjengelig for gjennomsyn på flere måter, bør søkemotoren i skjemakatalogen jobbes med. For å finne skjema gjennom søkefunksjonen i skjemakatalogen må man søke på spesifikke ord. Man kan kun søke på ord som finnes i tittelen på skjemaet. Man kan heller ikke sette sammen søk med forskjellige ord som et slags filter. Hvis jeg f.eks. ville ha alle skjema for næringsdrivende i 2004-utgave kunne jeg funnet på å søke på "næring 2004". Dette søket gir ingen treff – selv om det er flere skjema som inneholder både "næring" og "2004" i deler av tittelen.

Søk på termer som "selvangivelse" bør gi treff i aktuelle skjema for selvangivelsen, ikke bare de forskjellige selvangivelsesskjema. For eksempel bør man få opp det som i dag er kjent som vedleggsskjema til selvangivelsen ved søk på "selvangivelse".

### Lenking til skjema

En ting som har slått meg under mine undersøkelser av Altinn og skjemaene er at det finnes flere gode opplister av skjema utenfor Altinn. Flere av disse er bedre enn opplistingen vi finner i skjemakatalogen i Altinn. På skatteetaten.no finner vi en god opplisting av skjemaene til etaten. Denne peker til Skatteetatens PDF-skjemaer, altså papirkopiene.

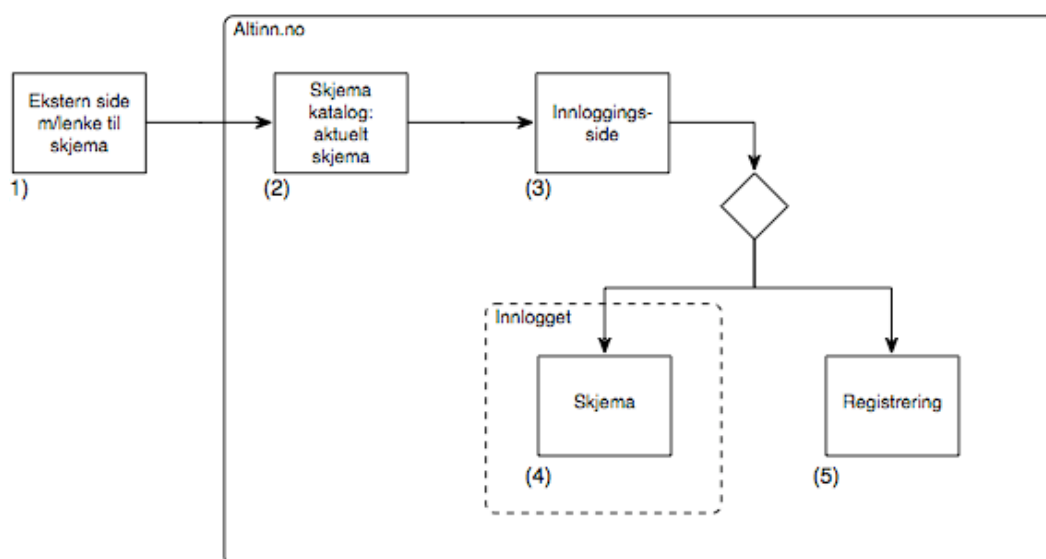
Likeledes opprettholder Altinn sentralforvaltning en opplisting av skjemaer i informasjonsportalen (utenfor sikkerhetsløsningen) med informasjon om de forskjellige skjemaene og hvem som skal fylle ut hvilke skjemaer.



Dagens Altinn er slik at skjemakatalogen ligger inne i sikkerhetsløsningen. Dette betyr at brukeren må være logget inn for å se skjemakatalogen. Et klart forbedringspotensiale i finnbarheten til skjema ville være å legge skjemakatalogen *utenfor* sikkerhetsløsningen, slik at brukere kan finne skjema både før og etter at de er logget inn i Altinn.

Ved å åpne skjemakatalogen vil man kunne gjøre det mulig å lenke til enkeltskjema fra andre steder på nettet. Bare dette i seg selv vil øke finnbarheten til skjema i Altinn betraktelig. For eksempel vil brukere kunne benytte seg av skjemakatalogen på skatteetaten.no og derfra komme til Altinn.

I Figur 14 har jeg satt opp et scenario for hvordan navigasjonen kunne foregå. Dersom en bruker står på et eksternt nettsted (1), for eksempel skatteetaten.no, kan det lenkes direkte til et skjema. La oss bruke aksjonærregisteroppgaven som eksempel. Når brukeren da følger lenken (2) kommer han til skjemakatalogen og en informasjonsside om dette skjemaet. Brukeren kan her velge om han vil legge til skjemaet i sin arbeidsliste ("fyll ut skjema"). Når han nå klikker seg videre (3) blir han bedt om å enten logge seg inn, eller bestille det han trenger for å logge seg på. Dette tilsvarer innloggingsdialogen som finnes i dagens løsning. Dersom brukeren allerede har tilgang til Altinn vil han komme inn i aksjonærregisteroppgaven (4). Dersom han ikke har tilgang til Altinn kommer han til en side som hjelper han å få tilgang (5).



**Figur 14: Forslag til navigasjonsflyt ved ekstern lenking til skjema i Altinn**

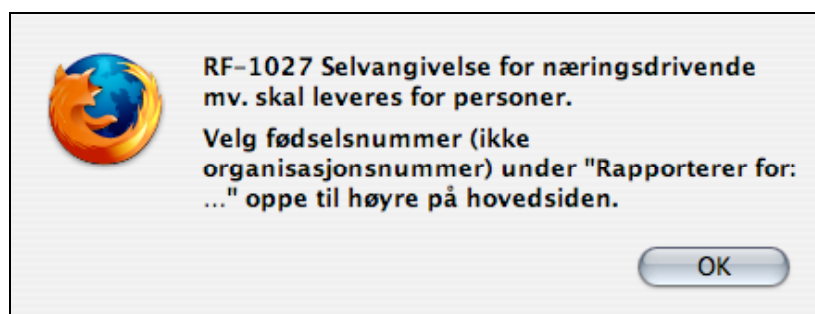
Code nettbutikker har operert på denne måten i mange år. Det har vært en forutsetning for å trekke kunder at brukerne kan surfe rundt i butikken, legge til varer i en handlevogn, og først *mot slutten* logge inn eller eventuelt registrere seg og dermed logge inn.

Forøvrig burde hele innloggings- og registreringsprosessen bli enklere når PKI kommer på plass i løsningen. Dersom brukeren da allerede er logget inn via PKI vil han komme rett til skjemaet fra skjemakatalogen.

Bruk av lenker er det mest sentrale aspektet av hypertekst. Dersom Altinn støttet lenking inn til tjenesten vil det gjøre tjenesten bedre. Muligheten til å sende direkte lenker til skjema vil også hjelpe på support. I stedet for at brukerstøtte må løse brukerne igjennom skjemakatalogen kan man sende en lenke til den aktuelle personen. Dersom brukeren allerede er i Altinn kan dette åpnes som et vindu og brukeren kan velge å legge inn skjemaet i arbeidslisten om han ønsker. Ikke minst kan folk lettere hjelpe hverandre ved å sende lenker til skjema direkte.

### Tilgjengelighet av skjema

Alle skjema er tilgjengelig for alle brukere (alle roller) etter innlogging. For eksempel kan en person som har enkeltmannsforetak gå inn som sin bedrift (på bedriftens organisasjonsnummer) og legge til selvangivelsen for næringsdrivende i sin arbeidsliste. Det er ikke før han velger å åpne skjemaet at han får en feilmelding om at dette skjemaet kun kan fylles ut av han selv som personlig skattyter (på sitt fødselsnummer). Hvis han da endrer rolle til seg selv, forsvinner skjemaet fra arbeidslisten. Hver rolle har nemlig sin egen arbeidsliste, brukeren må da legge til skjemaet på nytt.



**Figur 15: Dialogboks som kommer opp dersom en bruker prøver å åpne et skjema han ikke kan fylle ut.**

Hvorfor skal en bruker kunne legge til et skjema som han ikke får lov til å fylle ut? I stedet for at det kommer en feilmelding *etter* at han har lagt det til bør det heller være umulig for en bruker å legge til skjemaet i sin arbeidsliste. Feilmeldingen bør komme *før* han får lagt til dette skjemaet. Enda bedre hadde det vært dersom skjema som ikke kan fylles ut av brukeren er markert, slik at han skjønner at han ikke kan fylle ut skjemaet. Egentlig vil det si at det er den *rollen* brukeren innehar som ikke kan fylle ut skjemaet.

En annen løsning er å skjule de skjemaene brukeren ikke får lov til å fylle ut. Jeg tror dog at dette ikke er en god ide. Ufullstendige oversikter kan skape usikkerheter hos brukeren, spesielt for de som ofte bytter mellom roller. Det er bedre å vise alle skjema, men gjøre det klart hvilke skjema som brukeren kan fylle ut.

## Navigasjon og oversikt

### Venstremenyen gir ikke nok oversikt og navigasjonsstøtte

4. Venstremenyen består av korte overskrifter som kuttet. En del av plassen brukes også av tallene. Brukeren får kun hele overskriften ved å holde musepekeren over overskriftene. Dette gir dårlig oversikt over innholdet i skjemaet.

- Bryter med: 6) *brukerbyrde*, 9) *status*
- Alvorlighetsgrad: 4

Dette problemet er todelt. Dels skyldes dette hvordan menyen er implementert. Dels skyldes dette problemet de lange navnene på hver enkelt side. Sistnevnte er etatens ansvar. Her vil jeg diskutere implementasjonen av menyen fordi dette er noe som er satt av rammeverket og selve webportalen. I kapittel 6 vil jeg komme inn på navngivingen av sidene som en del av diskusjonen rundt utformingen av skjema og informasjonsarkitektur.

Et fundamentalt problem på nettet er å gi oversikt. Dette kalles gjerne "nøkkelhull-effekten" [Dieberger 1994]. På samme måte som når noen titter gjennom et nøkkelhull vil en bruker på internett bare se litt av det totale informasjonsrommet gjennom et skjermbilde. Brukeren vil i Altinn ikke ha oversikt over skjemaet på samme måte som et papirskjema i virkeligheten. For å bote på de iboende problemene med elektroniske medier må det utformes gode navigasjonssystemer som gir oversikt. I så måte har dagens venstremeny i Altinn et forbedringspotensiale (se Figur 16 på side 73 for skjermdumper av venstremenyen).

I følge Nielsen skal navigasjonssystemer besvare *tre* fundamentale spørsmål som brukeren har [Nielsen 1999a: 188]:

- Hvor er jeg?
- Hvor har jeg vært?
- Hvor kan jeg gå?

Venstremenyen er en del av et såkalt "embedded navigation system" [Rosenfeld og Morville 2002: 107]. Når menyen skal svare på så sentrale spørsmål som nevnt over er det viktig at menyen klarer å svare på disse spørsmålene så raskt og enkelt som mulig.

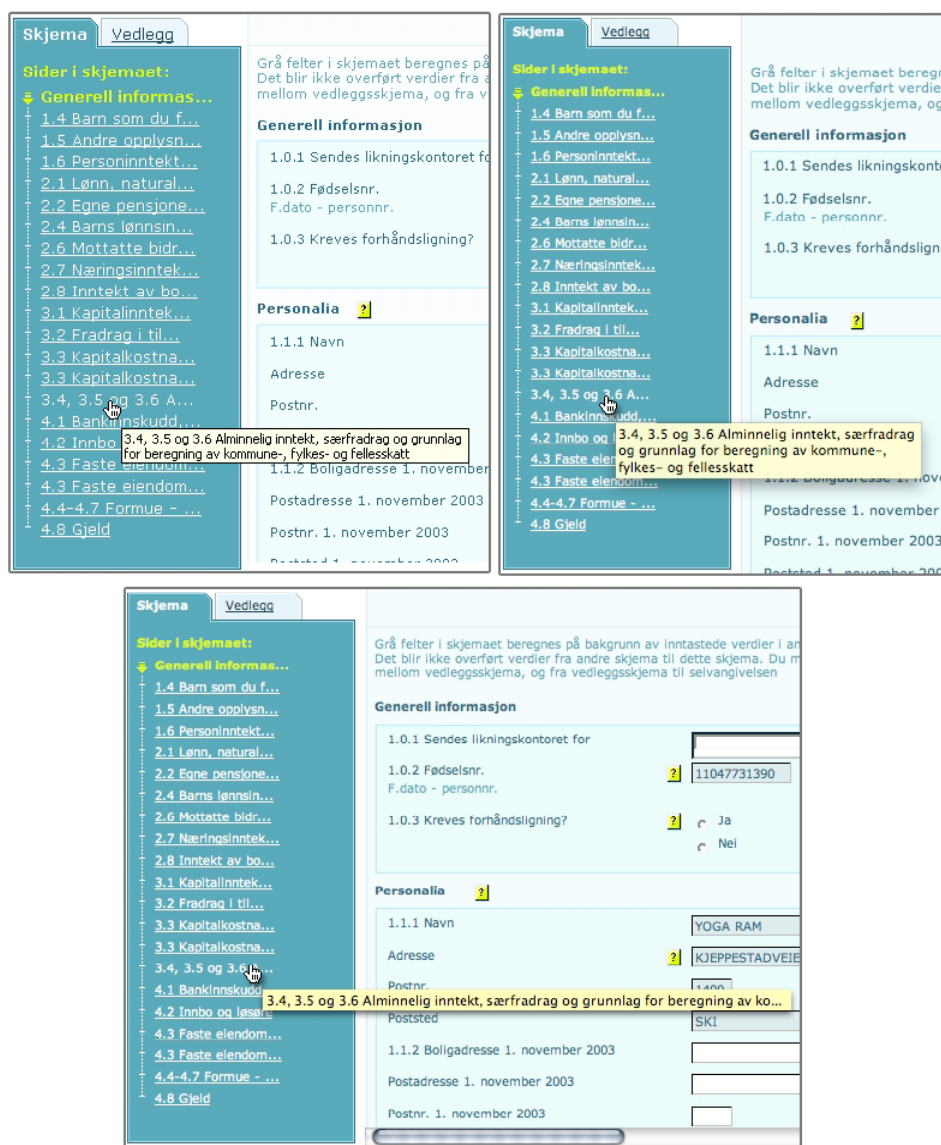
For å bote på at teksten i venstremenyen kuttet brukes det såkalte "title-attributter". Dette er en liten tekst som kommer opp ved siden av musepekeren når brukeren holder pekeren rolig over et HTML-element som også har title-attributt (se Figur 16). Fordi så mange av elementene i det testede skjemaet kuttet betyr det at menyen er ubrukelig for de som ikke har inngående kjennskap til skjemaets nummerering. Brukeren blir nødt til å holde pekeren rolig for å få (god nok) navigasjonsstøtte. Dermed må brukeren støtte seg til title-attributtene som primærverktøy for navigasjonsstøtte. Alternativet er å klikke seg til hver side uten å vite hva som

kommer, og dermed feiler navigasjonen på deler av hva den skal fungere til. Bruk av title-attributter bør *ikke* fungere som et primærverktøy for navigasjonsstøtte. Det er nemlig flere problemer med å være avhengig av denne teknikken.

For det første er informasjonen skjult helt til brukeren holder pekeren *rolig* over det aktuelle stedet. Dessuten må brukeren skjønne at det finnes mer informasjon bak det aktuelle elementet. Stort sett er det bilder og lenker som gir ut slike signaler, dette trenger altså ikke være et problem i dette tilfellet. I følge brukertester utført i regi av Accenture og Altinn sentralforvaltning viser det seg at mange av Altinns brukere ikke skjønner at venstremenyen er klikkbar.

Et annet problem er at implementasjonen av title-attributter varierer i nettlesere. Noen nettlesere, f.eks. Firefox *kutter* lange title-attributter. Andre nettlesere viser all teksten, men på en lang linje som er lite lesevennlig. Dersom teksten er veldig lang kan det hende at alt ikke vises dersom oppløsningen ikke er stor nok. Andre nettlesere igjen viser tekst som brytes og dermed gir lesevennlige "bobler".

Menyer som løper ned venstresiden slik som i Altinn har alltid liten plass til lange titler. I Altinns venstremeny synes kun 17 tegn (inklusive mellomrom) før overskriften kuttet med tre punktum.



**Figur 16: Sammenlikning av hvordan title-attributter vises i forskjellige nettlesere. Fra øverst til venstre - Internet Explorer 6 (Win), Safari 2.0 (Mac), Firefox 1.0.3 (Mac)**

Problemene i menyen skyldes dels den tekniske løsningen og begrensningene den legger, men også informasjonsarkitekturen og navngivingen til selve skjemaet. Den tekniske løsningen gjør at det ikke vil være hensiktsmessig å ha for mange sider (og dermed overskrifter) fordi venstremenyen ligger fast. Etatene kan heller ikke ha overskrifter som er lenge enn 17 tegn dersom de vil at overskriften skal være synlig og samtidig formidle nok informasjon. Dersom menyen blir lengre enn ett skjermbilde, blir det *scrollefeldt* ved venstremenyen også (se Figur 17).

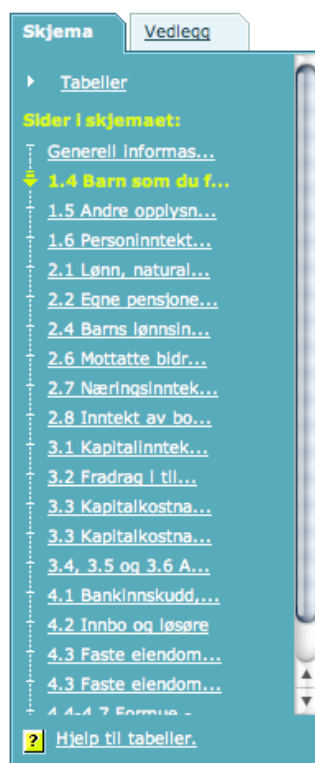
Jeg foreslår å fjerne kuttingen av overskriftene, i tillegg bør det åpnes for at lenkene kan løpe over flere linjer. Dermed fjernes de tekniske begrensningene og det blir opp til *etatene* å lage overskrifter som fungerer uten scrolling.

For at dette skal bli lettere for etatene bør det innføres to nye ting: mulighet for egne overskrifter i menyen, og mulighet for gruppering av overskrifter i menyen. I dag er overskriften på en side det samme som lenken i venstremenyen. I Skatteetatens skjema kan det være fordel å ha egne kortnavn i menyen, mens lengre navn brukes som overskrifter på siden.

Dersom det blir mulig å gruppere lenker vil det gi etatene mer frihet. Dermed kan overskriftene gi kontekst til lenkene som i dag blir gitt i lange overskrifter. Jeg kommer tilbake til dette i kapittel 6.

### Bedre utnyttelse av tilgjengelig skjermplass

Eksempelet i Figur 17 gir svært dårlig oversikt over skjemaet. Hvordan kan dette løses? Før vi går videre er det viktig å være klar over en ting. Å designe for nettet betyr at designeren må gi opp tanken på absolutt kontroll på hvordan noe ser ut. Forskjellige nettlesere, plattformer og skjermoppløsninger gjør at ting ser annerledes ut. Figur 16 viser for eksempel at tekststørrelsen i menyen er noe større på Internet Explorer på Windows enn i nettleserne i Mac OS.



Figur 17: Venstremeny med scrollefelt

I Figur 17 ser vi at venstremenyen er begrenset i begge akser. Det er kun 17 tegn til rådighet i bredden, og det er rundt 20 linjer i menyen før scrollefeltet dukker opp (dette tallet vil variere etter hvor stor skriftstørrelse nettleseren bruker). Hvis vi inntil videre ser bort fra selve navnene på de forskjellige sidene, kan vi si at den tekniske løsningen ikke bør være begrensende på navngivingen i menyen. Det vil være litt av en utfordring å lage gode merkelapper på de forskjellige sidene i skjemaet med kun 17 tegn til rådighet!

Å utvide menyen i bredden er ingen god ide. Det er blitt bestemt at løsningen skal fungere i 1024x768 skjermoppløsning. Ved å utvide venstremenyen vil vi enten spise av resten av innholdet (selve skjemaet og hjelpefeltet), eller vi vil introdusere horisontalt scrollefelt. Ingen av disse alternativene er gode, fordi plassen trengs av både innholdsfeltet og hjelpefeltet. Å introdusere horisontal scrolling er ingen god ide [Nielsen 2002]:

*"Web pages that require horizontal scrolling in standard-sized windows, such as 800x600 pixels, are particularly annoying."*

Horisontal scrolling alene er irriterende, men dersom det blir scrolling i begge akser er det mange som får problemer med å bruke systemet [Nielsen 2005]:

*"When pages feature both vertical and horizontal scrolling, users have to move their viewport in two dimensions, which makes it hard to cover the entire space. For people with poor spatial visualization skills, it's especially challenging to plan movements along two axes across an invisible plane."*

Hvis vi ikke kan utvide i bredden, kan vi prøve å utvide i høyden i stedet. Hvis vi lar menyen flyte sammen med skjemaet, i stedet for selvstendig som i dag, vil vi slippe begrensning på antall sider.

En annen mulighet ville vært å sette ned skriftstørrelsen i menyen. Dette er ikke en god ide, fordi teksten blir mindre lesbar. Teksten vil ved en nedjustering av størrelsen fortsatt være grei å lese (om ikke behagelig) for personer med godt syn. Men Altinn er en portal som skal henvende seg til hele befolkningen. Da er det viktig at det vanlige oppsettet har en tekststørrelse som er normalt lesbar. Det er også slik at de aller fleste "vanlige" brukere ikke vet hvordan de endrer tekststørrelsen til nettleseren. Så dersom teksten er for liten for dem vil de ikke vite hva de skal gjøre for å øke størrelsen.

Det vil fortsatt være svært viktig å holde antall elementer innenfor "folden" (folden er alt du ser innenfor ett skjermbilde uten å scrolle). Dette henger sammen med å gi brukeren oversikt [Nielsen 1999a: 196]:

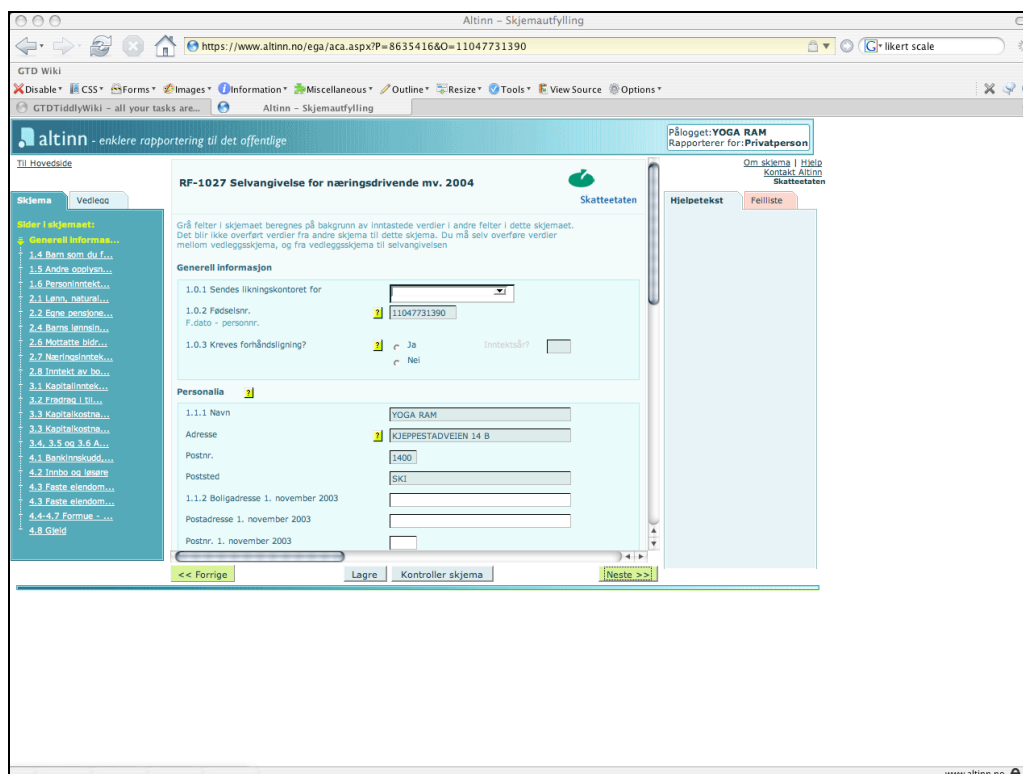
*"A navigation interface needs to show all the available alternatives at the same time so that users can make an informed decision as to which option will satisfy their needs best."*

En av de store fordelene ved å la menyen flyte sammen med resten av innholdet vil være at det kun blir ett felles scrollefelt for både innholdet i skjemaet og for navigasjonen. I de tilfellene det da *blir* scrolling på siden er det langt mindre brukerfiendtlig å ha *ett* scrollefelt i stedet for *tre*.

Det er ikke bare positivt å la menyen flyte med skjemaet. Den største fordelen med dagens løsning er at skjemaoversikten alltid er der – den ligger fast selv om brukeren

scroller ned i skjemasiden. Dersom den hadde vært mer informativ ville den gitt kontinuerlig oversikt over sidene i skjemaet.

Dagens Altinn løsning er bygget for oppløsning på 1024x768 punkter. Dersom brukeren har større oppløsning blir ikke dette utnyttet – det blir luft rundt. Som vi ser av Figur 18 utnyttes skjermplassen ikke spesielt godt på høyere oppløsninger. Eksempelet viser Altinn i et maksimert vindu på skjermoppløsning med 1280x1024 punkter.



**Figur 18: Altinn i maksimert vindu i skjermoppløsning 1280x1024 punkter. Legg merke til luften rundt innholdet.**

Å lage løsninger som skalerer godt etter forskjellige skjermoppløsninger er vanskelig med dagens standard-støtte i nettlesere. Det er problematisk å skalere opp bredden på løsningen, fordi etter som løsningen blir bredere og bredere mister vi lesbarheten i teksten (dette er en av grunnene til at papiraviser bruker smale kolonner).

Men ulempen ved å låse utnyttelsen av skjermplass i begge akser som i Altinn er at brukerne ikke får utnyttet den ekstra plassen. Dette er plass som kunne blitt brukt til å gi mer oversikt over skjemaet. Satt litt på spissen kan vi at jo mer brukerne ser, jo mer oversikt over skjemaet har de.

Slik Altinn er bygget i dag er det annen ulempe. Hvis det blir mye innhold i noen av de tre kolonnene – venstremenyen, skjemafeltet eller hjelpetekstfeltet, blir det scrollefelter i *hvert* av disse feltene (se Figur 19).



The screenshot shows the Altinn web portal interface. At the top, it says "altinn - enklere rapportering til det offentlige". The user is logged in as "YOGA RAM" and is reporting for "Privatperson". The main form is titled "RF-1027 Selvangivelse for næringsdrivende mv. 2004". On the left, there is a sidebar menu with a "Skjema" tab and a "Vedlegg" tab. The "Skjema" tab is active, showing a list of sections: "Tabeller", "Sider i skjemaet:", "Generell Informas...", "1.4 Barn som du f...", "1.5 Andre opplys...", "1.6 Personinntekt...", "2.1 Lønn, natural...", "2.2 Ekte pensjon...", "2.4 Barns lønnsin...", "2.6 Mottatte bidr...", "2.7 Næringsinntek...", "2.8 Inntekt av bo...", "3.1 Kapitalinntek...", "3.2 Fradrag i til...", "3.3 Kapitalkostna...", "3.4, 3.5 og 3.6 A...", "4.1 Bankinnskudd...", "4.2 Innbo og løsere", "4.3 Faste eiendom...", "4.4 Faste eiendom...", and "Hjelp til tabeller...". The "2.6 Mottatte bidr..." section is selected. The main form area shows sections 2.6, 2.6.1, 2.6.2, 2.6.3, and 2.6.4. Each section has input fields for "Utbetalers navn", "Beløp", and "Barnets f.nr.". On the right, there is a "Hjelpetekst" panel with a "Feilliste" tab. The "Hjelpetekst" panel contains text explaining the sections and providing instructions.

**Figur 19: Altinn i dag med scrollefelter i venstremenyen, skjemafeltet og i hjelpefeltet. Mer at det horisontale scrollefeltet skyldes Firefox sin tolkning HTML/CSS-koden, det er ikke pga. innholdet.**

Disse scrollefeltene dukker altså opp uavhengig av skjermoppløsningen til brukeren. Dette er ikke bare irriterende for brukeren, det gir dårlig oversikt.

Løsningen på dette er å gjøre det mulig å utnytte plassen i høyden. Vi kan la bredden være låst, og dermed spare oss for problemer med for brede kolonner og senket lesbarhet. Etter hvert som standardstøtten i nettleserne øker vil vi kunne innføre bruk av CSS-attributter som "min-width" og "max-width" for å få større kontroll på layout. Disse attributtene gjør det mulig å spesifisere minimum og maksimum bredder, med mulighet for skalering inntil en øvre eller nedre grense. Inntil CSS2-støtten blir bedre i nettleserne er dette noe som må vente.

Inntil da kan vi fjerne sperren i høyden, og la venstremenyen og hjelpefeltet flyte sammen med skjemaet. Dette vil redusere scrollefeltene i situasjoner som vist i Figur 19 fra tre til *ett* scrollefelt. Dessuten vil brukere som har høyere skjermoppløsning få inn mer informasjon innenfor "folden". Brukere med lavere oppløsning vil også nyte godt av dette – også de vil få kun ett scrollefelt i stedet for tre. Poenget er å få løsningen til å bli mer *uavhengig* av skjermoppløsning. I Figur 20 ser vi hvordan dette kan se ut.

Ulempen med denne løsningen er at både venstremenyen og hjelpeteksten ikke vil ligge fast. Det betyr at etter som brukeren scroller nedover på en side, vil menyen og hjelpeteksten "forsvinne" oppover. For hjelpetekst er ikke dette like farlig, dersom man gjør noen justeringer for visningen av hjelpeteksten. Jeg kommer tilbake til dette senere.

Jeg mener fordelene ved å la menyen flyte sammen med skjemainnholdet er større enn ulempene. Løsningen jeg skisserer skalerer bedre over forskjellige skjermoppløsninger, gir mulighet for å åpne for menynavn som kan gå over to linjer, og vi kan bli kvitt flere scrollefelter.

- 78 -

### Finne vedleggsskjema

5. Selv om man *vet* hvilket skjema man skal fylle ut er det vanskelig å finne det dersom det er et underskjema til et overordnet skjema som selvangivelsen. Å søke på "næringsoppgave" fungerer f.eks. ikke. Brukeren må legge til det overordnede skjemaet, og deretter (via hovedskjemaet) legge til de aktuelle vedleggsskjema.

- Bryter med: 4) *brukerkontroll*, 2) *finnbarhet*
- Alvorlighetsgrad: 4

Organiseringen av skjemaene i hoved- og vedleggsskjema (eller såkalte skjemasett) er en måte å organisere på som er etatssentrert. *Etaten* skal ha inn selvangivelsen. Alle vedleggsskjema brukes enten som hjelp til utfyller for å komme fram til tall som skal plottes inn i selvangivelsen eller annet hovedskjema, eller det brukes som dokumentasjon.

Mange selvstendig næringsdrivende starter ikke utfyllingen med selvangivelsen, men med *næringsoppgaven* og deretter personinntektsskjemaet. Disse brukes som grunnlag for å fylle ut selvangivelsen. Slik Altinn er lagt opp må brukeren først legge til selvangivelsen i sin arbeidsliste, gå inn i selvangivelsen og *derfra* legge til næringsoppgave og personinntektsskjema. Først *da* kan disse brukerne begynne å fylle ut de skjemaene de vanligvis begynner med.

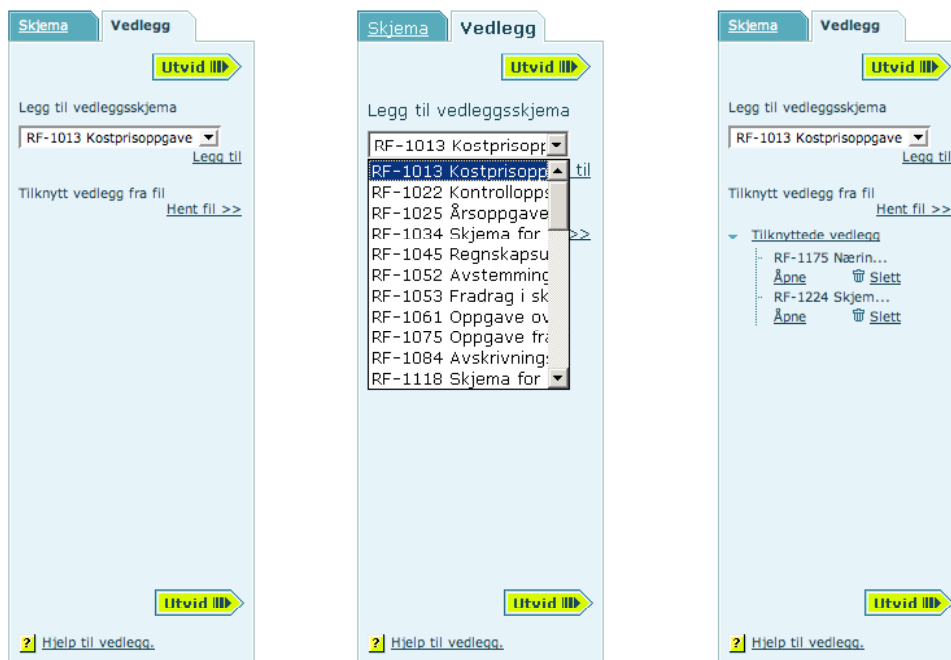
Dette strider med den *mentale modellen* [Norman 2002: 189] mange sporadiske brukere har av skjemautfyllingsprosessen. Her er det uoverensstemmelse mellom systembildet og brukerens modell [ibid: 190].

### Vedlegg og ekstra tabeller er ikke lett å finne før du har lært

6. Vedlegg og ekstra tabeller er ikke lett å finne (før du har lært det).

- Bryter med: 2) *finnbarhet*, 1) *konsistens*
- Alvorlighetsgrad: 3

Først og fremst er det vanskelig for nye brukere å *finne* vedlegg og tabeller. For å legge til vedleggsskjema må brukeren først ha lagt til og åpnet et hovedskjema. Deretter må han inn i "Vedlegg"-fanen i venstremenyen (se Figur 21). Da kommer det opp en fane med en liste med alle vedlegg som kan legges til i dette skjemaet. Som det kan ses av Figur 21 er lista liten og uoversiktlig i Internet Explorer (midterste bilde). Det er derfor de grønne "Utvid" knappene er der. Da utvides lista til å bruke hele skjermbredden. Brukeren kan så legge til de skjema han trenger – forutsatt at han finner de i lista. Lista er sortert etter RF-nummer, og det er ingen mulighet til å søke etter skjema ved å oppgi skjemanavn. I bildet til høyre har jeg lagt til to vedleggsskjema: Næringsoppgave 1 og Skjema for beregning av personinntekt.



**Figur 21: Oversikt over hvordan vedleggsskjema legges til i dagens Altinn. Vedleggsskjema administreres fra bildet til høyre.**

Det er flere problemer med denne tilnærmingen for å legge til vedleggsskjema. Det ene er problemet med den mentale modellen som jeg nevnte over. For mange sporadiske brukere er dette en unaturlig rekkefølge. Som en følge av det kan det være vanskelig for dem å skjønne hvilke skjema som finnes i skjemakatalogen, og hvilke som ikke finnes. Hvis de ikke skjønner den underliggende modellen for sortering, vil de ha problemer med dette.

Et annet problem er i forhold til hva slags informasjonssøkende oppførsel brukeren har. Hvis brukeren *ikke vet* hvilket skjema han skal legge til er det vanskelig å finne frem til riktig vedleggsskjema. Det finnes nemlig ingen søkemuligheter, det er kun nedtrekkslista brukeren kan titte gjennom.

Dersom brukeren *vet* hvilket skjema han skal legge til er det fortsatt problematisk å finne frem til riktig skjema. Problemet ligger i å bruke en nedtrekksliste av denne typen (HTML selectbox). Slike lister er vanskelig å scanne [Krug 2000: 115]:

*"[Pull downs are] hard to scan. Designers have no control over the font, spacing, or formatting of the list to make them more readable [...]"*

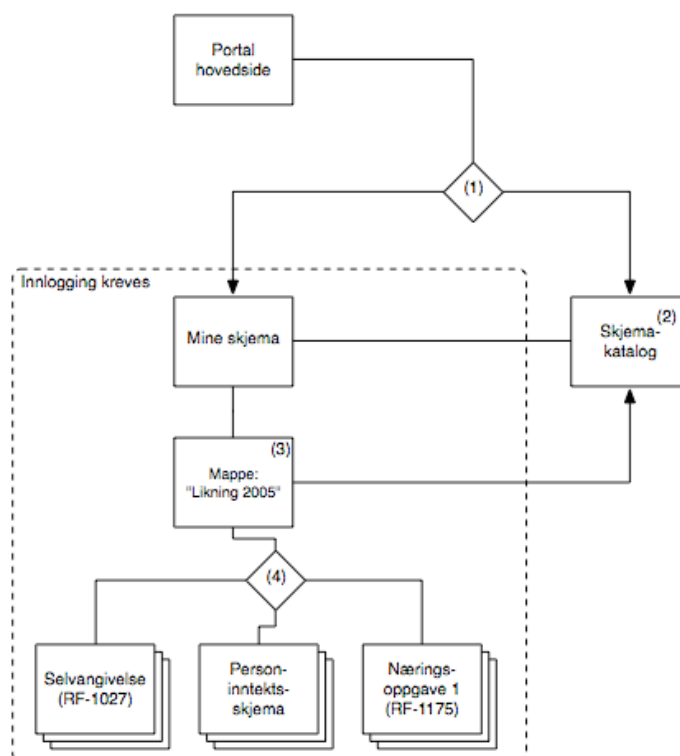
Nedtrekkslister egner seg best til lister med klar organisering der brukeren kjenner innholdet godt nok til ikke å måtte tenke særlig, men bare kan velge det han er ute etter. Et typisk eksempel er en liste over poststeder eller land. Når brukeren ikke er helt sikker på hva han er ute etter egner nedtrekkslister seg ikke like godt [Krug 2000: 115]:

*"[Pulldowns are] much less effective for lists where I don't know the name of the thing I'm looking for, especially if the list isn't alphabetized or is long enough to require scrolling."*

Nedtrekkslista er sortert etter RF-nummer. For de brukerne som husker RF-nummer på rams er dette en svært praktisk sorteringsmåte. Hvis brukeren ikke kan RF-nummeret derimot er det vanskelig å finne fram i lista.

I tillegg til å være vanskelig å finne, må vedleggsskjemaene administreres via et annet skjema, og det i et lite avgrenset område som venstremenyen. Hvorfor skal brukeren måtte administrere potensielt mange skjema via ett annet?

Mitt forslag er å kutte ut dagens oppsett av skjemasett der man har et overordnet skjema med en gruppe underskjema som vedlegg. I stedet kan man heller ha skjemasett som er en gruppe skjema samlet i en slags "mappe". Skjemaene vil da fremstå som likestilte for brukeren, selv om de ikke trenger være det for etatene. Dette passer bedre med den mentale modellen til sporadiske brukere.



**Figur 22: Forslag til bedre håndtering av vedleggsskjema: Innføring av mapper for skjemaadministrasjon**

Forklaring til Figur 22: Fra portalthovedsiden kan brukeren velge (1) om han skal gå til sine skjema (krever innlogging), eller gå til skjemakatalogen (2). Skjemakatalogen krever *ikke* innlogging, men er tilgjengelig også etter innlogging. I figuren har brukeren en mappe for "Likning 2005" (3). En mappe er en samling med relaterte skjema og han kan ha flere mapper dersom det er nødvendig. Alle skjema som hører til samme forsendelse må ligge i samme mappe. Fra mappen kan brukeren velge

hvilket skjema (4) han vil fylle ut. All skjemaadministrasjon i forhold til det som kalles vedleggsskjema i dag gjøres altså fra mappa, og ikke fra et overordnet skjema.

Dette innebærer introduksjonen av nok et nivå i hierarkiet i Altinn-strukturen. Man skal være forsiktig med innføre for mange nivåer, men jeg mener at fordelene ved å innføre noe slik veier opp for ulempene. Dessuten er det *egentlig* ikke snakk om å innføre et nytt nivå. Mappen vil ligge på *samme* nivå som dagens hovedskjema (for eksempel selvangivelsen).

Å lage en slik struktur innebærer to fordeler. Vi får da muligheten til å lage et dedikert grensesnitt for å håndtere de skjemaene som skal leveres i forbindelse med et skjemasett. Et eget skjermbilde kan brukes til å legge til, slette og kontrollere skjema. Dette vil være langt bedre enn dagens løsning med skjemaadministrasjon i en trang venstremeny inne i et annet skjema. Vi kan kutte ut et modus i hovedskjemaet, og vi får frigjort plass i venstremenyen som heller kan brukes til å bedre venstremenyen og navigasjonen (og dermed øke oversikten over skjemaet). Dette er bra fordi skjemaet vil være enklere og mer fokusert på utfylling. I tillegg vil det å flytte deler av funksjonaliteten ut fra skjemaet bidra til å forenkle skjermbildet i skjemaet.

Den andre fordelene ved å løfte opp vedleggsskjemaene et nivå er at det vil gjøre det enklere for brukerne å finne vedleggsskjemaene. Dette forutsetter at disse skjemaene nå finnes i skjemakatalogen på lik linje med dagens hovedskjema og er søkbare.

Den største ulempen med innføring av slike mapper vil være dersom det kun skal leveres ett skjema til en etat. I de tilfellene der etaten *vet* at det kun skal leveres ett skjema kan mappen droppes.

### Vedleggstabeller

7. Når man legger til instanser i en vedleggstabell (f.eks. arbeidsgivere) og går tilbake til selve skjemaet er det vanskelig å se at tallene der har blitt oppdatert. Kun ved utregning av sum-feltet i den aktuelle posten vet man at dataene har blitt lagret og overført til hovedskjemaet. Dette gjør brukeren usikker.

- Bryter med: 6) *brukerbyrde*, 3) *gode dialoger*, 9) *status*
- Alvorlighetsgrad: 3

**Skjema** Vedlegg

**Tabeller**

**Sider i skjemaet:**

- 1.4 Barn som du f...
- 1.5 Andre opplys...
- 1.6 Personinntekt...
- 2.1 Lønn, natural...**
- 2.2 Egne pensjone...
- 2.4 Barns lønnsin...
- 2.6 Mottatte bidr...
- 2.7 Næringsinntek...
- 2.8 Inntekt av bo...
- 3.1 Kapitalinntek...
- 3.2 Fradrag i til...
- 3.3 Kapitalkostna...

**2.1 Lønn, naturalytelser mv.**

Alle beløp avrundes ned til nærmeste hele krone

2.1.1 Lønn honorarer og annen godtgjørelse

Lønn, honorarer og annen godtgjørelse. Sykepenger, attføring- og rehabiliteringspenger, tidsbegrenset uførestønad og dagpenger under arbeidsledighet til lønsmottakere. Naturalytelser, f.eks. fri bil, se rettledningen.

Arbeidsgiver/utbetaler	Tidsrom	2.1.1 Beløp
Skattedirektoratet		300000

Hvis du vil registrere flere, velg rett tabell fra menyen Tabeller som du finner til venstre.

Sum post 2.1.1 skattepliktig beløp = 1650000

**Figur 23: Bruk av vedleggstabell for arbeidsgivere. Det er to arbeidsgivere spesifisert i vedleggstabellen som skjuler seg bak lenken "Tabeller" (markert med pil, t.v.). Den eneste indikasjonen på dette i hovedskjemaet er i sum-feltet til posten (markert med pil t.h.).**

Det eneste brukeren kan se er at *summen* for den aktuelle posten er blitt oppdatert. Dette er en svært diskret statusendring. En klar løsning på dette problemet er å få dynamiske tabeller inn i løsningen der brukeren kan legge til så mange elementer han vil.

Det er også et annet problem, og det er å finne *igjen* vedleggstabeller. Lenken til tabeller kommer *kun* opp på de sidene der det er mulig å legge inn vedleggstabeller (se Figur 23 for hvordan lenken ser ut). Hvis brukeren vil gjøre endringer i tabellvedlegg må han altså huske hvilke sider tabellene befinner seg på.

Bruken av vedleggstabeller henger igjen etter papirutgaven av skjemaet. På papir er det naturlig nok begrenset med plass, og dersom vi har behov for ytterligere plass må dette legges ved. Dette er et problem som ikke finnes i samme grad i elektroniske skjema. I essens har vi uendelig med skjermplass, så lenge vi godtar scrolling. Dynamiske tabeller er noe som har stått høyt på ønskelisten til etatene. Dette er noe som vil bli mulig å legge inn i skjema når Altinn oppgraderer til neste versjon av skjemamotoren.

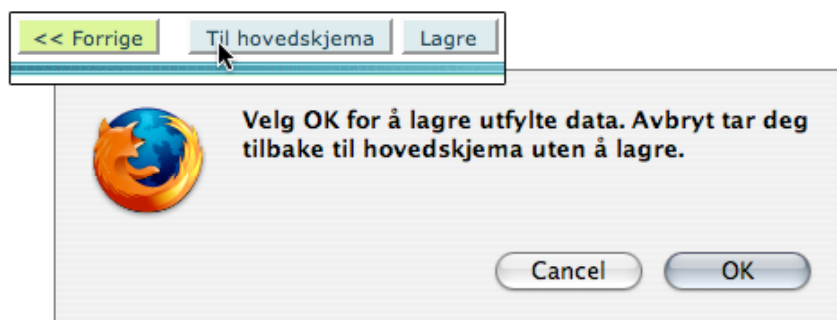
## Lagring av brukers arbeid

### Inkonsistente dialogbokser ved navigering

- Spørsmål om lagring ved navigering fra vedlegg- til hovedskjema har motsatt logikk i forhold til samme situasjon for navigering fra hovedskjema til hovedside.
  - Bryter med: 1) *konsistens*
  - Alvorlighetsgrad: 3



Figur 24 : Dialogboks ved klikk på "Hovedside" fra et skjema.



Figur 25: Dialogboks ved klikk på "Til hovedskjema" fra vedleggsskjema.

Dette er et punkt som gjør at brukeren alltid må stoppe opp ved dialogboksen og forsikre seg om at han gjør riktig valg. Det er mulig dette er gjort med hensikt, nettopp for at brukeren *ikke* skal tillegge seg en vane og alltid klikke på samme sted. På den annen side, er det i det hele tatt er nødvendig å spørre om skjemaet skal lagres? *Alle* sider lagres ved navigering ellers i skjemaet, hvorfor skal brukeren da få muligheten til å velge om den aktuelle siden skal lagres når han skal til hovedsiden? Slik dialogboksene er må brukeren nå bruke kognitiv prosessering på å finne ut hva dialogen sier, i stedet for å kunne svare på den kun via gjenkjenning.

Om det nå er slik at det skal være en dialogboks er det et spørsmål om den skal ha et så potensielt destruktivt valg. De fleste vil velge å lagre det de har gjort. Men fordi logikken mellom de to nevnte dialogboksene er motsatt er det stor risiko for at brukeren en gang klikker en annen knapp enn den han mente. Dette vil innebære at brukeren mister alle data siden forrige lagring eller siden forrige gang han navigerte mellom sider.

#### Bruk av dialogbokser ved avslutning

9. Man blir eksplisitt bedt om å lagre når man går ut av skjema. Hvis man ikke lagrer eller klikker seg videre blir det man har fylt ut ikke lagret.
  - Bryter med: 8) ivareta arbeid
  - Alvorlighetsgrad: 3



I Figur 24 ser vi hvordan dialogen er når en bruker klikker seg til hovedsiden. Hvorfor er ikke det vanlige valget å lagre endringene? Når en bruker først har klikket på "Hovedside"-lenken er han innstilt på å komme tilbake til forsiden. En slik dialogboks kan raskt oppfattes som en hindring på veien til målet og derfor "avfeies" ved å klikke på "OK" uten videre tanker. I så fall vil brukeren miste endringene på den siden han var på. Sannsynligheten for at dette kan skje øker også fordi dialogboksene i skjema og vedleggsskjema har motsatt logikk (se punktet over).



**Figur 26: Løsningsforslag for dialogboks ved klikk på "Hovedside".**

I Figur 26 har jeg skissert opp et løsningsforslag som vil være en bedre løsning, og som bygger på følgende retningslinje for design [Preece, Rogers og Sharp 2002: 83]:

*"Design interfaces that promote **recognition** rather than **recall** by using menus, icons, and consistently placed objects."*

Utgangspunktet bør være at en bruker alltid ønsker å lagre det som er fylt ut i et skjema. Følgelig bør systemet alltid ta vare på informasjonen. Men det kan finnes tilfeller der brukeren *ikke* ønsker å lagre. Da er det viktig å tilby brukeren muligheter for å angre det han gjør; knappen til hovedsiden kan tilby dette. Ved å bygge om dialogen slik jeg har foreslått blir det vanligste valget lettere tilgjengelig via gjenkjenning, *samtidig* som det tilbyr angre-muligheten som brukeren kan trenge.

### Trøbbel ved timeout

10. Man står fortsatt i skjema-bildet etter automatisk utlogging (timeout). Man kan altså potensielt fortsette fylle ut resten av siden, for så å trykke på "neste". Først *da* får man beskjed om at man må logge inn, og det man har gjort på foregående side blir borte.
  - Bryter med: 4) *brukerkontroll*, 8) *ivareta arbeid*, 9) *status*
  - Alvorlighetsgrad: 4

Et av problemene her er at brukeren ikke aner hvor lang tid det tar før han mister kontakten med tjeneren. Dette er sannsynligvis et problem som er større for de av oss som kjenner til web-teknologien og dens begrensninger. De som ikke kjenner til dette problemet vil være lykkelig uvitende om problemet. Altinn tar heldigvis høyde for dette problemet. Man blir heller ikke logget ut av Altinn før det har gått en god stund uten aktivitet (rundt 30 minutter). Problemet er de brukerne som vet at timeout kan være et problem.

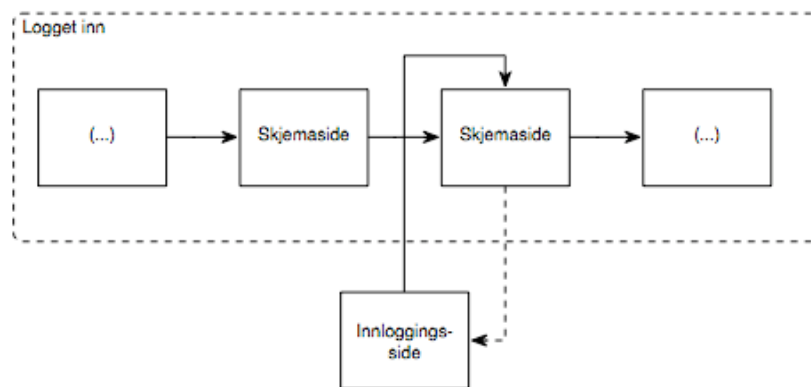
Timeout mot serveren var et problem som blant annet Lånekassen slet med tidligere. I deres elektroniske nettsøknad var de fleste sider raske å fylle ut. Men enkelte sider tar lengre tid å fylle ut. Hvis vi f.eks. er forsinket i utdannelsen krever Lånekassen at vi skriver inn en forklaring på hvorfor vi er forsinket. Dette kan ofte kreve en del tid å skrive. Det er flere som har brukt lang tid på å skrive en ordentlig forklaring for så å miste den når de trykker på "Gå videre"-knappen fordi nettleseren har mistet sesjonen med tjeneren. Lånekassen har løst dette ved å øke vinduet mot serveren til 20 minutter, og ved å legge inn en teller som vises på nettsiden slik at brukeren har en indikasjon på hvor lang tid han har før tidsbegrensningen er nådd.

**Figur 27: Skjermdump fra Lånekassens nettsøknad. Legg merke til telleren med gjenstående tid (markert med pil).**

Vi snakker egentlig om to problemer her. Det ene er å gi god indikasjon på hvor lang tid du har du før du mister forbindelsen til tjeneren. Det andre er å si klart og tydelig ifra når du først *har* mistet forbindelsen. Slik Altinn er satt opp i dag har du ingen indikasjon på hvor lang tid du har igjen. Men desto verre; du ser ut som om du fortsatt er i skjemaet, selv etter du har mistet forbindelsen.

Det er kanskje ikke nødvendig å opplyse om hvor lang tid brukeren har igjen. Det har nemlig ikke bare fordeler. Å legge inn en slik teller kan også stresse brukeren fordi det minner han på hvor lite tid som er igjen. Dessuten er det vanskelig å kommunisere akkurat hva telleren egentlig skal *formidle*. Har brukeren x minutter igjen på å fylle ut *hele skjemaet* eller gjelder tallet kun *denne siden*? –Hva skjer når telleren kommer til null? For *brukeren* bør det være likegyldig hvor lang tid han har til rådighet – det er en systembegrensning som vi designere og utviklere bør håndtere.

Ved en eventuell timeout bør systemet lagre skjemaet der brukeren er. Deretter bør innloggingssiden lastes. På innloggingssiden bør det stå klart og tydelig at brukeren har blitt logget ut etter x antall minutter uten aktivitet, og at dette er blitt gjort av sikkerhetshensyn e.l. Når nå brukeren logger inn på nytt bør systemet hente inn den siden brukeren var på før han ble logget ut automatisk. Dermed kan han enkelt ta opp igjen arbeidet sitt.



Figur 28: Løsningsforslag til navigasjonsflyt ved timeout

### Lagres skjema ved klikk i menyen?

11. Venstremenyen er navigerbar, men gir ikke gode nok signaler om data blir lagret. (Er vant til statiske menyer på nettsteder der forms *ikke* blir lagret ved klikk på meny).

- Bryter med: *affordances (ny)*
- Alvorlighetsgrad: 3

Evaluatorene i den heuristiske evalueringen skjønte at man kan bruke venstremenyen til navigasjon, men det var enkelte som var usikre på om data ble lagret ved klikk i venstremenyen. Dette har sammenheng med at vi befinner oss på nettet, der de fleste vanlige nettsteder ikke lagrer det du taster på en side dersom du klikker i menyen. Altinn er en webapplikasjon og sliter dermed litt med etablerte måter å jobbe på. De mest brukte webapplikasjonene på nettet ellers, som webmail og nettbanker har sjelden dialoger som går over flere skjermbilder. De gangene de har det, brukes ikke noen meny ved siden, kun "gå videre"-knapper i umiddelbar tilknytning til feltene man har endret.

Selv om ekspertene som gikk gjennom løsningen skjønte hvordan menyen fungerte, er det mange vanlige brukere som ikke skjønner at man kan bruke venstremenyen til navigasjon. Brukertester som er foretatt av Altinn sentralforvaltning og Accenture forteller at mange brukere ikke bruker venstremenyen til navigasjon. Mange tror den bare er der for fortelle brukeren hvor han er, men at han må bruke forrige/neste-knappene for navigere.

Relatert til dette er usikkerhet blant evaluatorene om hvorvidt man måtte trykke på "lagre" før man klikker seg videre via "neste"-knappen.

Denne usikkerheten på om skjemaet lagres ved navigasjon og om man må klikke på "lagre" for hver side er uheldig. Problemet her er at Altinn fremstår som en mellomting mellom et nettsted og en webapplikasjon. Denne usikkerheten vil nok

bare gjøre seg gjeldende for nye brukere. Like fullt er det en usikkerhet som ikke bør være der.

En mulig løsning er å legge inn en liten tekst over menyen der vi forklarer at menyen kan brukes til navigasjon og at data blir lagret ved bruk av menyen. Det som er problematisk med dette forslaget er at det allerede er lite plass i venstremenyen. En slik tekst vil stjele verdifull plass som kan brukes til navigasjonslenker i selve menyen.

Dette problemet henger sammen med den sentralt plasserte lagre-knappen. Jeg diskuterer lagre-knappen under.

### Bruk av lagreknapp

12. Lagreknapp er sentralt plassert, men det er uklart hvor ofte den må brukes eller om andre funksjoner lagrer i bakgrunnen.

- Bryter med: 8) *ivareta arbeid*
- Alvorlighetsgrad: 3

Lagre-problematikk er noe som alle web-applikasjoner støter på. De fleste web-applikasjoner har sjelden en egen lagre-knapp. F.eks. har de fleste webmail-klienter og nettbanker ikke knapper som er navngitt "lagre", men "ok", "legg til" osv. Disse applikasjonene er mer aksjonsorienterte.

Det fine med å ha en synlig lagre-knapp i Altinn er at det gir brukeren et klart signal om at han bare trenger å klikke på knappen for å lagre eller ta en pause.

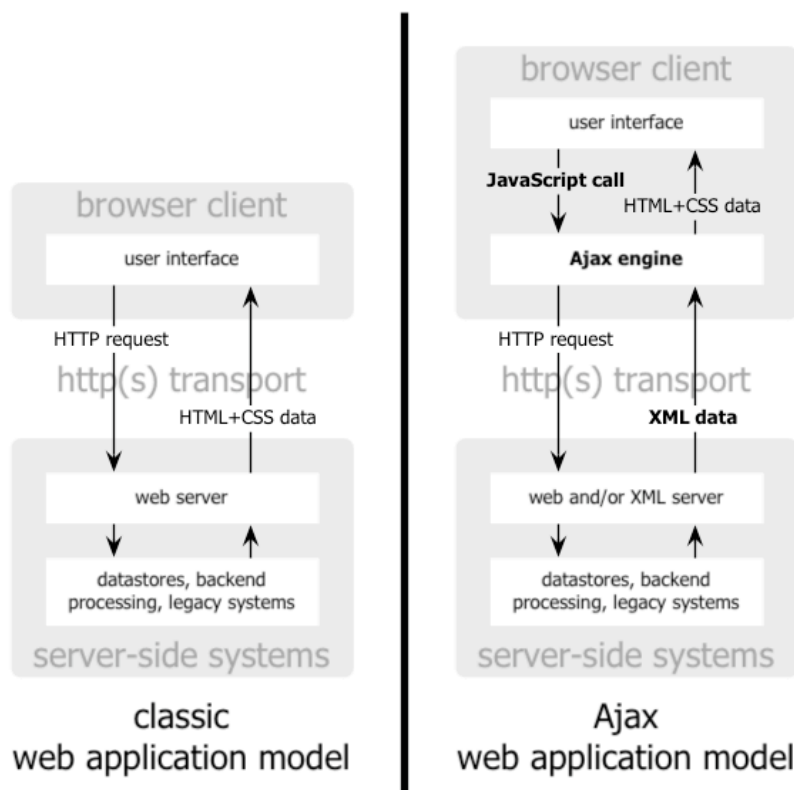
Problemet med en så synlig lagre-knapp er at det gjør brukeren usikker på om systemet lagrer skjemaet ved klikk på menyvalgene i venstremenyen. En av evaluatorene lurte på om han måtte klikke "lagre" for deretter å klikke på "neste" eller i menyen. Bruk av en eksplisitt lagre-knapp skaper altså usikkerhet om det er trygt å bruke andre valg for å navigere.

Spørsmålet er om vi trenger en eksplisitt lagre-knapp? Hvis menyvalg og forrige / neste-knapper også lagrer, har vi virkelig behov for en egen knapp for lagring?

Problemet med å fjerne lagre-knappen er at brukeren ikke lenger har en enkel måte å lagre arbeidet på uten å måtte navigere fram og tilbake. Det kan f.eks. hende at brukeren bruker lang tid på en side og vil mellomlagre. Dette *kan* løses ved at systemet lagrer automatisk. Men dette må da gjøres *uten* at siden laster på nytt og dermed forstyrrer brukeren.

Dette kan løses ved å bruke en såkalt Ajax-løsning [Garrett 2005]. Ajax står for "Asynchronous JavaScript + XML" og er en kombinasjon av eksisterende web-teknologier for å lage rikere webapplikasjoner. Den største fordelene med bruken av Ajax er at utviklerne ikke trenger å ta hensyn til den tradisjonelle klient-tjener

modellen, trafikk kan gå mellom klienten og tjeneren når det trengs *uten* å laste nettsiden på nytt.



**Figur 29: Oversikt over forskjellen mellom tradisjonell web-modell og Ajax-modell**

Ved å bruke Ajax-modellen kan Altinn lagre sømløst i bakgrunnen, uten at brukeren merker det.

I motsetning til mange web-applikasjoner skal Altinn støtte muligheten til å la brukeren legge fra seg arbeidet for så å kunne ta det opp igjen senere ved anledning. Dette er en av grunnene til at lagre-knappen er så sentralt plassert. Men dersom systemet lagrer i bakgrunnen vil det ikke være det samme behovet for en lagre-knapp.

Mitt forslag er å introdusere lagring ved hjelp av Ajax. Deretter foreslår jeg å døpe om dagens lagre-knapp til "Pause". Ved klikk på denne vil skjemaet lagres og brukeren vil deretter logges ut. Her ser jeg for meg tilsvarende løsning som for timeout som jeg skisserte opp tidligere. Brukeren vil logges ut og det vil stå en passende forklaring om at skjemaet er lagret og at brukeren kan logge inn igjen for å fortsette arbeidet. Når brukeren da logger inn igjen vil han bli tatt til samme side som han var på før han trykket på pause.

Dette forslaget har to fordeler. Det ene er at det tilfredsstiller behovet for å legge fra seg skjemaet. Det andre er at det eliminerer usikkerheten om lagre-knappen må brukes – det vil ikke finnes noen lagre-knapp.

## Hjelpesystemet

Forbedring av hjelpesystemet er todelt. Den ene delen dreier seg om å tilrettelegge og forbedre hjelpesystemet i rammeverket, slik at det bedre kan støtte brukeren. Den andre delen dreier seg om bedre *innhold* i hjelpetekster. Sistnevnte skal jeg prate mer om i kapittel 6. Her vil jeg ta for meg forbedringspotensialet i hjelpesystemet.

## Altinns hjelpemuligheter

13. Det er mange liknende valg når man søker etter hjelp: "hjelp", "hjelpetekst", "kontakt Altinn" og "Om skjema". Det er ikke opplagt for brukeren hvor han skal henvende seg for hva.

- Bryter med: 6) *brukerbyrde*
- Alvorlighetsgrad: 3

Altinn har fra skjermbildet i skjemautfylling fire valg som gir pekepinn om hjelp. For nye brukere vil det ikke være klart hvor han skal henvende seg for hva. "Fasiten" er som følger:

- "Hjelp"  
Bak dette valget finner brukeren hjelp til å bruke selve *verktøyet* Altinn. Det er altså hjelpefunksjonen til programvaren, eller brukermanualen. Her finner brukeren informasjon om hvordan utfyllingsprosessen er lagt opp, hvordan han finner skjema, hvordan han bruker vedleggstabeller m.m.
- "Hjelpetekst"  
Dette er ment å inneholde fokusert hjelpetekst som skal hjelpe brukeren i utfyllingsøyeblikket. Dette er tatt fra ELMER.
- "Kontakt Altinn"  
Dette er teknisk brukerstøtte. Her får brukeren svar på problemer han skulle ha med selve løsningen. Han får *ikke* hjelp til utfylling av skjema her. Det er det den enkelte etat som håndterer.
- "Om skjema"  
Her finnes informasjon om skjemaet – hvilket etat som eier skjemaet, hvem som skal fylle det ut osv. Dette er generell informasjon om selve skjemaet.

Det at Altinn tilby hele fire forskjellige valg i en hjelpesituasjon påvirker flere scenarier. Jeg kommer inn på dette i diskusjonene rundt de andre punktene under.

## Finne rettledningen

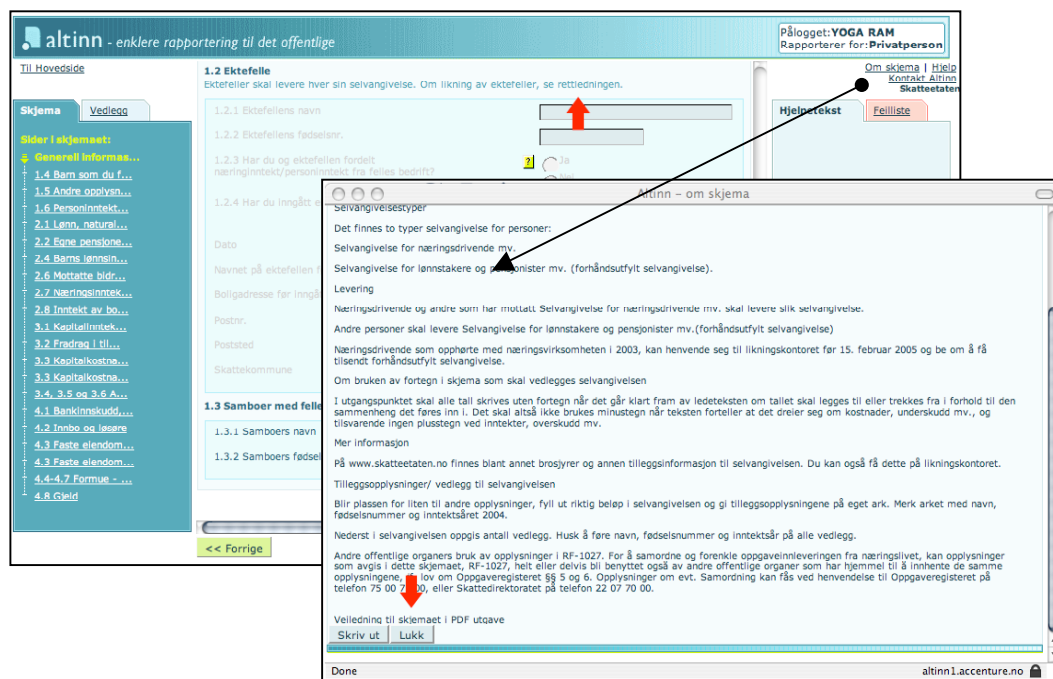
14. Det står flere steder at man kan se i rettledningen dersom man har spørsmål. Men det å finne rettledningen er ikke enkelt.

- Bryter med: 6) *brukerbyrde*, 2) *finnbarhet*, 12) *hjelp*
- Alvorlighetsgrad: 4

Hjelpeteksten som finnes tilgjengelig i Altinn er for Skatteetatens del for det meste en oppstykket papirrettledning. Hvis en bruker vil ha tak i selve papirrettledningen

finnes den tilgjengelig. Men den er ikke enkel å finne. I Figur 30 ser vi at det står en henvisning til rettledningen (markert med pil i det bakerste vinduet):

*”Om likning av ektefeller, se rettledningen.”*



**Figur 30: For å finne rettledningen må brukeren inn på "Om skema" siden og finne lenken helt nederst på siden.**

Hvis brukeren ikke vet *hva* rettledningen er, kan det hende at han prøver å sjekke de forskjellige valgene han har til rådighet for hjelp. Det er ikke innlysende å vite at "rettledningen" er *papirutgaven* av rettledningen til *papirskjemaet* som ligger tilgjengelig et sted som PDF. Hvis en bruker leter etter hjelp er det "Hjelp"-lenken som er den mest sannsynlige kandidaten. Det er ikke noe spørsmålstegn-ikon ved siden av den nevnte teksten, det er dermed ikke noen hjelpetekst til feltet. "Hjelp" leder til hjelpesystemet for selve løsningen, og ikke skjemaet.

Hvis brukeren klikker på "Om skema"-lenken kommer han inn på en side med generell informasjon om selvangivelsen. Denne siden blir intetsigende for en som leter etter hjelp. Det blir heller ikke bedre av at siden ikke støtter "scanning" [Nielsen 1999a: 104, Krug 2000: 22] ved at teksten er uten formatering (overskrifter er ikke markert med fet skrift e.l.). Helt nederst på siden ligger det en diskret lenke til PDF-utgaven av rettledningen.

Legg også merke til at lenken til PDF-utgaven heter "Veiledning til skemaet i PDF-utgave". Ellers i skjemaet og hjelpetekst kalles veiledningen for *rettledning*.

Ut i fra samtaler med flere brukere vet jeg at mange har lett etter rettledningen på Altinn og ikke funnet den. Deretter har de gått til Skatteetaten.no og funnet fram til rettledningen der, for så å komme tilbake til Altinn for å fylle ut skjemaet. Slik bør det ikke være!

Papirveiledningen er et fint supplement til selvvalgt hjelpetekst. Det å ha muligheten til å skrive ut hele rettledningen og ha den ved siden av datamaskinen er det mange som gjør. At papirrettledningen er tilgjengelig er derfor bra. Men rettledningen bør være enklere å finne – spesielt når den refereres til i skjemaet. Hvorfor det må refereres er et annet spørsmål. Informasjonen burde heller ha blitt lagt inn som selvvalgt hjelpetekst.

I de tilfeller der det *er* nødvendig å referere til rettledningen kan det legges inn lenker direkte til rettledningen. Dermed slipper brukeren å lete etter den. I eksemplet vi har gjennomgått kunne dette sett slik ut:

### 1.2 Ektefeller

Ektefeller skal levere hver sin selvangivelse. Om likning av ektefeller, se [rettledningen \(PDF 2,1 MB\)](#).

Her lenkes det direkte til rettledningen. Samtidig oppgir vi for brukeren at det er en PDF-fil (og ikke en webside) som det lenkes til, og vi oppgir filstørrelsen slik at brukeren kan belage seg på at dette kan ta litt tid å laste ned. Det er utstrakt bruk av PDF i Altinn, men det står sjelden eller aldri hvor store disse filene er. For brukere som ikke har bredbånd er dette svært irriterende. Filstørrelser på lenker til store filer som PDF eller store bilder bør alltid oppgis for å informere om at dette vil ta tid.

I ELMER-demoen [Nygaard 2001b] var det lenket til rettledningen direkte i skjemabildet (se Figur 4 på side 16). Etter min mening tror jeg ikke rettledningen trenger å løftes så høyt – hvertfall ikke dersom selvvalgt hjelpetekst er lagt inn. Rettledningen bør heller være en sekundær hjelp, og trenger følgelig ikke løftes frem like mye. Dessuten er det allerede fire valg som minner om hjelp. Dette er forvirrende for brukeren.

Mitt forslag er å legge rettledningen *øverst* på "Om skjema"-siden. I tillegg bør den utheves visuelt slik at den klart skiller seg ut fra brødteksten.

### Kontaktinformasjon

15. Det er vanskelig å finne kontaktinformasjon til faglig hjelp for utfylling av skjema.

- Bryter med: 6) *brukerbyrde*, 4) *brukerkontroll*
- Alvorlighetsgrad: 3



Det at Altinn har så mange forskjellige valg der brukeren kan henvende seg for hjelp bidrar til å gjøre det vanskelig for brukeren å finne den hjelpen han trenger. De mange valgene henger sammen med måten Altinn er organisert på – med Altinn sentralforvaltning som forvalter selve løsningen og portalen, mens etatene har ansvaret for det enkelte skjema og dets tilhørende hjelpetekst.

Siden det er to forskjellige instanser som opprettholder de forskjellige funksjonene blir det problematisk for brukeren å vite hvor han skal henvende seg. Hvis brukeren f.eks. har et spørsmål til utfylling av skjemaet er det ikke opplagt hvilken av lenkene som kan hjelpe han. Hvis hjelpeteksten ikke er til hjelp kan han prøve å klikke på "Hjelp". Dette er hjelp til portalen, ikke til skjemaet. Hvis brukeren da bestemmer seg for å *spørre* om hjelp er det bare ett valg som sier noe å kontakte noen: "Kontakt Altinn".

"Information foraging" [Pirolli og Card 1999] er en teori om hvordan brukere leter etter informasjon i elektroniske media. Teorien bruker analogien om at mennesker er som ville dyr på jakt etter mat for å analysere hvordan vi samler informasjon på nettet. En av grunnpilarene er at folk er *late* – de orker ikke å anstrenge seg for mye for å finne informasjon. [Nielsen 2003]:

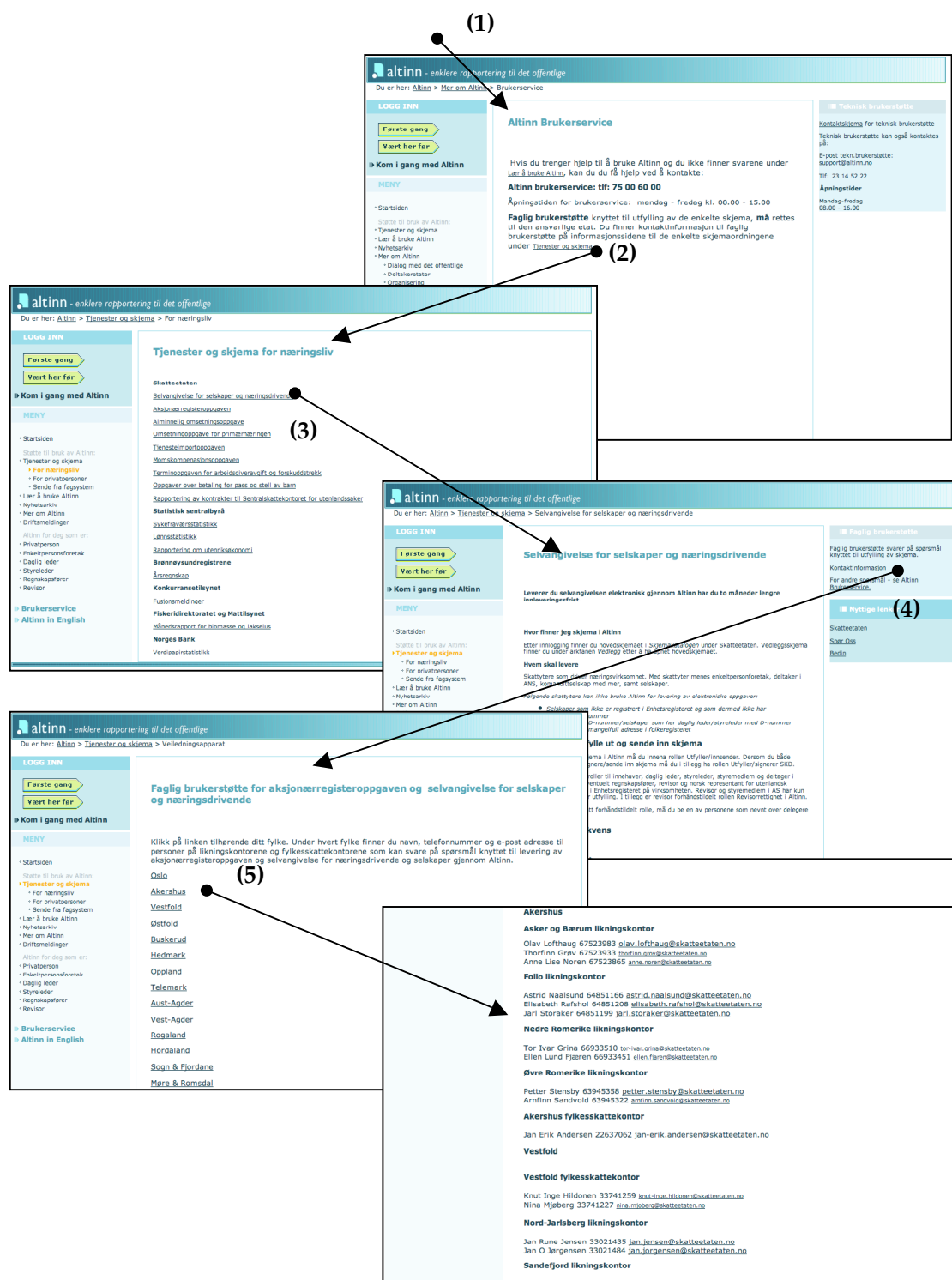
*"[...] people like to get maximum benefit for minimum effort. That's what makes information foraging a useful tool for analyzing online media."*

Det mest kjente konseptet fra denne teorien er "information scent" – brukere vil kun følge lenker som de tror vil bidra til at de kommer til "byttet". Eller som Nielsen sier om information scent [Nielsen 2003]:

*"[...] users estimate a given hunt's likely success from the spoor: assessing whether their path exhibits cues related to the desired outcome."*

Hvis en lenke ikke er lovende (gir fra seg riktig "lukt") leter de et annet sted. Brukere på jakt etter informasjon av et slag vil fortsette å klikke så lenge de føler at de er på sporet av informasjonen de er ute etter. For at brukerne ikke skal gi opp må de føle at de nærmer seg "byttet", ellers vil de gi opp raskt, eller klikke seg tilbake for å se om det går an å gå en annen vei til informasjonen. Vi som designere må derfor lage gode navn på lenker som beskriver det som finnes seg bak dem. Vi må også passe på å bruke vanlig språk som gir ifra seg riktig "lukt".

Figur 31 tar for seg navigasjonen for å finne fram til kontaktinformasjonen. Brukeren må gjennom fem sider før han ender opp med den informasjonen han trenger for å ta kontakt med faglig brukerstøtte (dette gjelder for Skatteetaten). Forutsatt at han ikke gir opp før dette. Under følger en tekstlig gjennomgang av navigasjonen for å finne fram til kontaktinformasjonen (se Figur 31 for skjermdumper og navigasjonsflyt).



Figur 31: Navigasjonsflyt for å finne hjelp til utfylling av selvangivelsen for næringsdrivende. Tallene er "klikk-nummer", dvs rekkefølgen brukeren følger. (Skjermdumper tatt 8/7-05)

(1) Bruker klikker på "Kontakt Altinn" fra skjemaet. Han lander da på en side som heter "Altinn brukerservice". Her står blant annet følgende beskjed:

*"Faglig brukerstøtte knyttet til utfylling av de enkelte skjema, må rettes til den ansvarlige etat. Du finner kontaktinformasjon til faglig brukerstøtte på informasjonssidene til de enkelte skjemaordningene under Tjenester og skjema."*

Bruker klikker på linken "Tjenester og skjema" (2) og kommer til "Tjenester og skjema for næringsliv". Rekkefølgen på etater og lenker her er litt merkelig, men for vårt tilfelle er det utmerket at selvangivelsen ligger øverst. Bruker klikker seg inn på "selvangivelse for selskaper og næringsdrivende" (3).

Bruker kommer nå inn på en side som heter "Selvangivelse for selskap og næringsdrivende". Han leter fortsatt etter *kontaktinformasjon*. Men denne siden ser ikke ut til å inneholde noe kontaktinformasjon – det ser snarere ut som en informasjonsside om skjemaet. Her vil mange brukere gi opp, fordi siden ikke gir noe *umiddelbart svar* på hva brukeren i denne situasjonen leter etter. I en liten diskret boks på høyresiden står overskriften "Faglig brukerstøtte" med lenken "Kontaktinformasjon" (4).

Nå blir det straks mer lovende. Overskriften "Faglig brukerstøtte for aksjonærregisteroppgaven og selvangivelse for selskaper og næringsdrivende" er lang, men "faglig brukerstøtte" lover godt. Det er litt forvirrende at bruker må velge hvor han bor for å få hjelp. Bruker kan så klikke på det fylket han bor i (5).

Brukeren ender så til slutt på en oversikt over likningskontor i fylket han bor i, med direkte telefonnummer til ansatte på likningskontoret. Dette forutsetter at brukeren vet hvilket likningskontor som er saksbehandlende enhet for hans selvangivelse.

Som vi så av gjennomgangen var det mange muligheter underveis for at brukeren ville gi opp. Det er ikke lett å finne riktig kontaktinformasjon her. Mange av brukerne ville nok heller ikke skjont forskjellen mellom Altinn og Skatteetaten, og ville nok derfor ringt til Altinn support (altså teknisk support). De ville derfra sikkert blitt overført til riktig avdeling, eller til Skatteetaten. Men det er både i brukernes og etatenes interesse at brukere som trenger hjelp kommer til riktig sted så fort som mulig.

Siden hjelp til utfylling er såpass viktig er løsningsforslag enkelt. Kontaktinformasjon til etaten burde løftes opp på samme nivå som resten av hjelpen, og ligge tilgjengelig i skjemaet.

Ulempen er at dette da medfører *fem* alternativer for hjelp i skjemaet. En løsning på dette er da å endre ordlyden på "Kontakt Altinn" til "Kontakt oss". På siden under "Kontakt oss" bør da både kontaktinformasjon til Altinn teknisk support ligge, og kontaktinformasjonen til etaten som eier skjemaet. Her er det viktig å løfte opp den informasjonen brukeren er etter så høyt som mulig. Siden systemet kan få tak i hvilket skjema brukeren fyller ut kan vi vise informasjonen i (5) på Figur 31 med en gang.

## **Veiviser**

16. Veiviseren i portalen kom ikke opp i det hele tatt i Firefox (både Mac og Windows) og Safari (Mac).

- Bryter med: 11) *feilavverging*, 12) *hjelp*
- Alvorlighetsgrad: 3

Med tanke på at Altinn i skrivende stund skal støtte de fleste moderne nettlesere (se punktet om nettleserstøtte på side 13) er det ikke bra at Altinns introduksjonsveiviser ikke kommer opp i det hele tatt i enkelte nettlesere som løsningen skal støtte. Dette burde kontrolleres og utbedres.

## **Formatering av hjelpetekst**

I dag er det ingen formateringsmuligheter i hjelpeteksten. Det betyr at etatene har svært få virkemidler til å gjøre teksten mer lesevennlig. For å gjøre teksten mer vennlig for lesing, og enklere for scanning burde det bli mulig å legge inn full HTML-formatering i hjelpetekstene. Ikke bare fet skrift, men mulighet for å angi avsnitt, overskrifter, lister og lenker også. Nielsen angir noen råd for skrive for web [Nielsen 1999a: 101]:

*“Write for scannability. Don’t require users to read long continuous blocks of text; instead use short paragraphs, subheadings, and bulleted lists.”*

Å kunne lenke til annen hjelpetekst vil også være et viktig hjelpemiddel for å øke verdien av hjelpetekstene.

## **Fler-lags hjelpetekst**

Mye av dagens hjelpetekst er vanskelig å forstå. Dette har sammenheng med at fagavdelingene i etatene er opptatt av at hjelpeteksten holder juridisk standard – det skal være korrekt. For *brukerne* betyr dette gjerne at hjelpeteksten blir mindre lesevennlig. Ofte vil brukerne ha behov for en enkel innføring i begreper. For å tilfredsstille begge parter kan det innføres fler-lags hjelpetekst. Dette er hjelp i to “nivåer” der brukeren først får se en “enkel” tekst som er optimalisert for utfyllingsøyeblikket. Fra denne teksten kan det lenkes til en mer fagrettet tekst, gjerne med lenker til Lovdata for å tilfredsstille fagavdelingene og de profesjonelle utfyllerne.

## **Bedre markering av sammenheng mellom felt og valgt hjelpetekst**

I dagens Altinn er det vanskelig å se hvilket felt som har markert hjelpetekst. Dette vil vanligvis ikke være noe problem i det brukeren trykker.

Jeg foreslo tidligere at venstremenyen og hjelpeteksten bør flyte sammen med skjemaet. Dersom vi beholder dagens løsning på hjelpeteksten vil det bety at

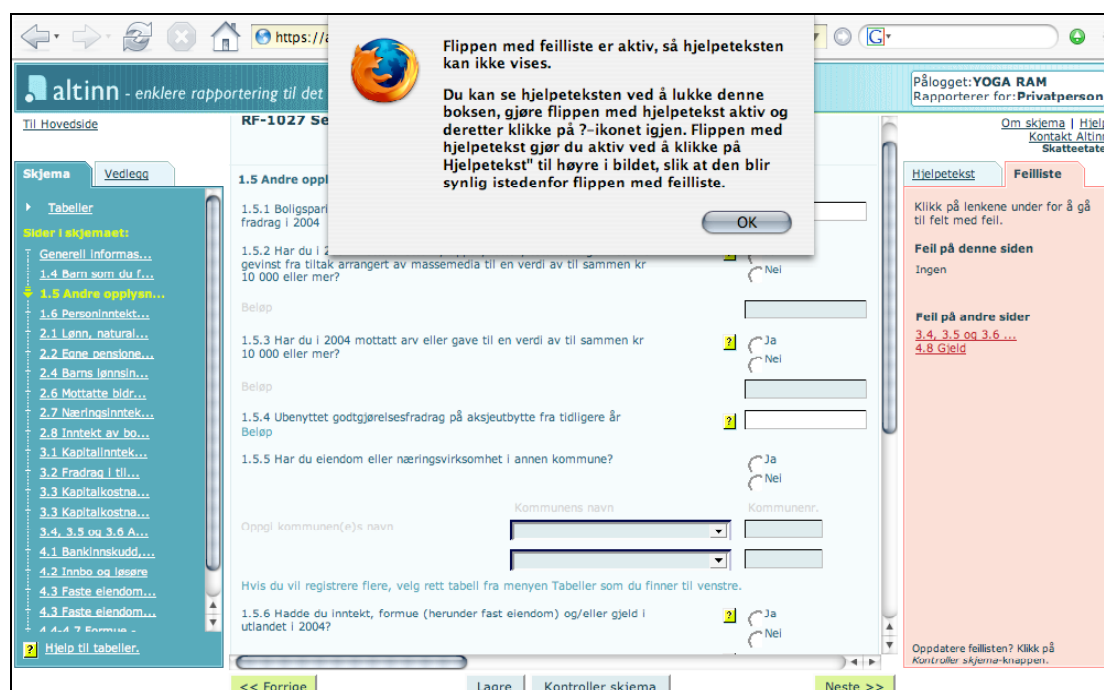
brukeren ikke kan se hjelpeteksten og det aktuelle feltet samtidig. For å bøte på det foreslår jeg å innføre at hjelpeteksten kommer i samme høyde som feltet du ber om hjelp til. Dette tilsvarer forslaget som ble fremsatt av ELMER-gruppa [Nygaard 2001b: 11].

## Feilretting

### Ingen hjelp ved feilsøk

17. Når man retter en feil og har feilflippen åpen går det ikke an å klikke på hjelpeikoner. Det kommer da opp en dialogboks som ber deg klikke frem hjelpeflippen manuelt.

- Bryter med: 4) *brukerkontroll*, 3) *gode dialoger*
- Alvorlighetsgrad: 3



Figur 32: Feilflippen er åpen, men brukeren kan ikke få hjelp

Dette er svært irriterende oppførsel fra systemets side. Når brukeren er innstilt på å rette feil er det naturlig å trykke på hjelpeikonene ved de aktuelle feltene, slik at han kan få mer informasjon om hva som kreves fylt ut i det aktuelle feltet. At brukeren potensielt må gjennom *fire* klikk (tre hvis han husker hva han må gjøre) er langt i fra smidig. Først klikker brukeren på spørsmålstegn-ikonet. Deretter får han opp dialogboksen med beskjeden. Han klikker da "ok" i dialogboksen. Deretter må han klikke på hjelpetekst-fanen, og så kan han klikke på spørsmålstegn-ikonet igjen.

Det bør være mulig for brukeren å klikke direkte på spørsmålstegn-ikonet og dermed få hjelp. Når han så vil se feilene igjen bør han kunne klikke på feilliste-fanen. Det bør være enkelt å bytte mellom feil- og hjelpetekst-fanen.

## Skjult forklaring på feil

18. I feillisten etter skjemakontroll står utdypende forklaring til feilene i mouseover-tekst. Denne vises ikke lenge nok til at man får lest hele feilmeldingen. Følgelig må man flytte pekeren frem og tilbake flere ganger for å få med seg feilen. Den er heller ikke synlig før man holder musepeker rolig over linken, alle vil derfor ikke finne den.

- Bryter med: 10) *feilhåndtering*, 3) *gode dialoger*, 9) *status*
- Alvorlighetsgrad: 3



**Figur 33: Feilene listes opp i høyre kolonne. Legg merke til at feilmeldingene blir kuttet, og den komplette feilmeldingen kun er tilgjengelig ved å holde musepekeren over lenken.**

Dette er ikke alltid et problem. Problemet oppstår som regel når navnet på siden er langt. Altinn vil da kutte tittelen (slik det gjøres i venstremenyen). Brukeren vil da ikke kunne se hva feilen består i før han kommer inn på den aktuelle siden. Hvorfor det er blitt bestemt at teksten skal kuttes her kan man spekulere i. Sannsynligvis var det for å spare plass, slik at brukeren kunne få opp mange feil uten å scrolle. Men slik løsningen fungerer i dag gir Altinn for *lite* informasjon til brukeren.

Mitt forslag her er å vise mer informasjon i feillisten. Dette kan med fordel stå i fulltekst, slik at brukeren får et best mulig utgangspunkt før han velger hvilke feil han skal rette.

19. I feillisten etter skjemakontroll kommer detaljer om feilen opp som mouseover-tekst. Denne teksten blir i enkelte nettlesere forkortet, slik at man ikke får med seg meningsinnholdet. Dette skjer f.eks. i Firefox (Mac).

- Bryter med: 10) *feilhåndtering*
- Alvorlighetsgrad: 3

I Figur 33 ser vi at forklaringen til feilen kommer opp som en tekstboble (såkalt mouseover-tekst). Denne blir i Firefox forkortet slik at du ikke får med deg hele forklaringen. Også her blir løsningen å vise feilmeldingen i fulltekst, i stedet for korte den ned slik det gjøres i dag.

### Låste felter

20. Disabled-felter (ikke-skrivbare felter) er vanskelig å skille fra vanlige felter. Dette gjelder spesielt på skjermer med lavere kontrast (som f.eks. skjermer på bærbar).

- Bryter med: 9) *status*
- Alvorlighetsgrad: 3

Dette er et problem som kommer tydeligst frem under feilretting. Dersom du ser to felter ved siden av hverandre, der det ene er åpent og det andre låst ("disabled"-attributtet i HTML) er det ikke så vanskelig å skille mellom de.

Dersom feltet er fylt ut kan det være vanskelig å se om feltet er åpent eller ikke. Dette gjelder *spesielt* når skjemakontroll gjør oppmerksom på at feilen skyldes et gitt felt, der det gitte feltet blir markert. Hvis dette feltet f.eks. er et sum-felt vil det da være markert, men likevel låst.

Mitt forslag for å løse dette er å se på de visuelle virkemidlene som brukes. Det er mulig å gjøre noe med fargebruken og utseende til disse låste feltene for å gjøre det tydeligere at disse feltene ikke lar seg endre.

## 5.5. Oppsummering

I dette kapittelet har jeg sett på rammeverket og den overordnede løsningen til Altinn. Jeg har analysert funn fra en heuristisk og sett på løsningsforslag til disse funnene. Jeg har også tatt opp noen ideer jeg mener kan gjøre brukeropplevelsen i Altinn bedre.

Nedenfor følger en oppsummering av funnene jeg har behandlet, med tilhørende løsningsforslag. Enkelte funn har felles løsningsforslag. Dette fremgår av tabellen ved at cellen for løsningsforslaget dekker de aktuelle funn.

ID	Problem	Løsningsforslag
1	Skjemakatalogen er avsenderorientert, ikke brukerorientert.	Skjemakatalogen bør tilby alternative sorteringsmåter, for eksempel etter brukergrupper, emner og oppgaver.
2	Søk på tema eller innhold i skjema mangler.	Det bør gå an å søke på metadata til skjema, ikke bare i tittel.
3	Man må søke på svært spesifikke ord for å finne skjema i skjemakatalogen.	
4	Venstremenyen gir ikke nok oversikt og navigasjonsstøtte.	La venstremenyen flyte sammen med skjemaet. Åpne for at lenker kan gå over flere linjer og droppe kuttingen av teksten. Mulighet for å gruppere lenker og bruke overskrifter på disse gruppene av lenker.
5	Selv når brukeren vet hvilket	Innføring av "mapper", slik at dagens hovedskjema vises på lik linje med dagens

	skjema han skal fylle ut er det vanskelig å finne dersom det er et vedleggsskjema.	vedleggsskjema. I essens droppes hele vedleggsskjema-organiseringen. <i>Alle</i> skjema finnes i skjemakatalogen. Brukeren kan organisere alle skjema i en leveranse fra den aktuelle "mappa".
6	Vedlegg og ekstra tabeller er ikke lett å finne.	
7	Det er vanskelig å se oppdateringer i hovedskjema etter at man har lagt til data i vedleggstabeller.	Innføring av dynamiske tabeller. Bruk av dynamiske tabeller vil gjøre at vedleggstabellene kan kuttes fra løsningen.
8	Dialogbokser ved navigering fra skjema til hovedside har motsatt logikk i forhold til samme situasjon for navigering fra vedlegg til hovedskjema.	Gjøre logikken i dialogboksene like, slik at brukeren kan ta nytte av gjenkjenningseffekten og klikke på samme valg.
9	Man blir bedt om å lagre når man går ut av skjema. Hvis man er uforsiktig risikerer man å miste data.	Gjøre "ok" det vanlige valget som lagrer, slik at brukeren slipper den omstendelige prosessen ved å trykke "avbryt", "lagre", "avslutt".
10	Etter automatisk utlogging står man fortsatt i skjemabildet.	Systemet bør lagre rett før timeout. Deretter bør systemet vise innloggingssiden, med en tekst som forklarer hva som har skjedd. Ved innlogging vises siden brukeren var på før timeout.
11	Venstremenyen er navigerbar, men gir ikke gode nok signaler om data blir lagret.	Systemet kan lagre automatisk i bakgrunnen ved innføring av Ajax-teknikker. Lagre-knappen kan døpes om til "Pause" for å opprettholde behovet til brukeren for å legge fra seg skjemaet. Oppførselen til "pause"-knappen bør være lik som ved timeout (skissert tidligere).
12	Lagreknapp er sentralt plassert, men det er uklart hvor ofte den må brukes.	
13	Det er mange liknende valg når brukeren leter etter hjelp.	Fire valg beholdes, men med én endring i navnene på hjelpevalgene: "Hjelp", "Hjelpetekst", "Kontakt oss" og "Om skjema".
14	Det er vanskelig å finne rettledningen til skjema.	Selvvalgt hjelpetekst bør brukes til å besvare flest mulig spørsmål. I tilfeller der det er nødvendig å referere til rettledningen bør det lenkes direkte til den. Rettledningen kan ligge på "Om skjema"-siden, men bør ligge øverst og være visuelt uthevet fra brødteksten.
15	Det er vanskelig å finne kontaktinformasjon til faglig hjelp for utfylling av skjema.	Endre "Kontakt Altinn"-lenken til "Kontakt oss". På denne siden bør det ligge kontaktinformasjon til både Altinn teknisk brukerstøtte, og til faglig brukerstøtte.
16	Veiviser i informasjonsportalen kommer ikke opp i en del moderne nettlesere.	Hvorfor dette skjer bør bli avklart. Deretter bør det bli utbedret.
17	Når man retter en feil og har feilflippen åpen går det ikke an å klikke på hjelpeikoner.	Det bør gå an å klikke direkte på hjelpeikoner under feilretting uten at det kommer opp dialogbokser.
18	Detaljer om feil krever at brukeren flytter musepekeren fram og tilbake for å få med seg feilen.	La hele feilmeldingen stå i vanlig tekst, og ikke skjult som mouseover-tekst.
19	Detaljer om feil kommer opp som mouseover-tekst. Denne kuttes i enkelte nettlesere.	
20	Disabled-felter er vanskelig å skille fra vanlige felter.	Endre det visuelle utseende på låste felter slik at det blir lettere å se forskjell på vanlige felter og låste felter.



## 6. FORBEDRINGSPOTENSIALET I SKJEMAUTFORMING

### 6.1. Introduksjon

I kapittel 4.4 på side 45 så vi at brukeropplevelsen i Altinn er et todelt ansvar. Altinn sentralforvaltning har ansvaret for webportalen. Etatene har ansvaret for hvordan skjemaene deres er bygget opp – selve skjemaautforming. I kapittel 5 så vi på forbedringspotensialet i webportalen. I dette kapitlet skal vi se nærmere på forbedringspotensialet i skjemaautforming.

Alle skjemaer i Altinn har en informasjonsarkitektur; de består av sider, der hver side kan deles opp i logiske grupper. Alle sidene bør ha logiske navn og en fornuftig rekkefølge. Alt dette er elementer som informasjonsarkitektur handler om – organisering og strukturering av informasjon.

I dette kapitlet vil jeg se nærmere på oppbygningen av informasjonsstrukturen i ett skjema, nærmere bestemt selvangivelsen for næringsdrivende (RF-1027). Jeg vil ta for meg informasjonsarkitekturen i dette skjemaet og se på hvordan skjemaet kan bygges om slik at det vil fremstå som et bedre skjema for brukerne.

Først presenteres metodene som er blitt brukt, deretter resultatene. Til slutt diskuteres resultatene og løsningsforslag til de identifiserte problemene.

### **Retningslinjer for utforming av elektroniske skjema i Skattedirektoratet: eSkjema SKD**

Som en del av mitt arbeid med masteroppgaven har jeg sittet i flere prosjektgrupper i Skattedirektoratet (SKD). En av prosjektgruppene var "eSkjema SKD". Formålet til denne prosjektgruppa var å utarbeide retningslinjer for utforming av skjema innenfor dagens Altinn-løsning (versjon 3.0/3.1) som på best måte tilfredsstiller både sluttbrukere (næringsliv og privatpersoner) og interne ønsker og behov i Skatteetaten.

Arbeidet som ble gjort i denne prosjektgruppa innebar å hente inn erfaringer med skjema i Altinn (fra både sluttbrukere og etatsbrukere), analysere disse erfaringene og sette sammen et sett med retningslinjer for å lage gode elektroniske skjema. Mye av det jeg presenterer her er basert på erfaringer og innsikt jeg fikk i arbeidet med dette prosjektet.

## 6.2. Definisjon

Vi har allerede definert informasjonsarkitektur på et overordnet nivå (se kapittel 1.4). Litt mer konkret kan vi si at informasjonsarkitektur i hovedsak dreier seg om tre ting [Rosenfeld og Morville 2002: 5]:

- **Strukturering**  
dreier seg om å finne ut av størrelsen på informasjonsgrupper og elementer, og hvordan disse er relatert.
- **Organisering**  
dreier seg om å gruppere de ovennevnte gruppene inn i meningsfylte og distinktive kategorier.
- **Navngiving** (engelsk: labeling)  
dreier seg om å gi navn til kategoriene og linkene som binder kategoriene sammen.

Peter Morville opererer også med begrepet "findability" som er sterkt knyttet til informasjonsarkitektur. Som Morville og Rosenfeld sier [Rosenfeld og Morville 2002: 5]: *"If users can't find what they need through some combination of browsing, searching, and asking, then the site fails"*.

## 6.3. Metode

### Arbeidet i eSkjema SKD

Jeg vil her presentere metodene vi brukte i eSkjema-prosjektet. Arbeidet som ble gjort i denne prosjektgruppa resulterte i et sett med retningslinjer som skal implementeres i Skatteetatens elektroniske skjema for skatteåret 2005.

### Erfaringer med Altinn

Som en del av arbeidet i eSkjema SKD ble det samlet inn informasjon om brukererfaringer med Skattedirektoratets skjema i Altinn. Disse resultatene ble analysert av prosjektgruppa og presentert i en erfaringsrapport [Udjus 2005].

Funnene var delt opp i tre kategorier: generelle anbefalinger; struktur, layout og tekst; og tall og beregninger.

Alle disse funnene var ikke nødvendig for oss å se på. Tall og beregninger dreide seg stort sett om kalkuleringer i skjemaene, noe som ikke hadde fungert spesielt bra i Altinn til nå. Vi tok for oss erfaringene og delte dem opp i tre nye kategorier:

- 1) Konsistens
- 2) Struktur
- 3) Layout

Disse erfaringene ble så analysert. Basert på disse erfaringene og en ekspertevaluering av Altinn begynte vi å formulere retningslinjer for direktoratets elektroniske skjema i Altinn.

### **Prototyping**

For å teste potensielle løsninger og retningslinjer ble det konstruert en rekke prototyper med begrenset funksjonalitet. Disse prototypene ble utviklet i HTML og JavaScript og ble satt sammen for å illustrere forskjellige scenarioer og prinsipper.

Prototypene ble brukt på to måter. Først og fremst som et middel for å kommunisere og teste ut mulige løsninger på problemer internt i prosjektgruppa. Prototypene ble også testet på brukere utenfor prosjektgruppa for å teste at løsningsforslagene vi hadde faktisk fungerte.

### **Brukertesting/fokusgruppe**

For å teste løsningsforslagene vi hadde kommet opp med ble det holdt en brukerevaluering av prototypene vi hadde laget.

Testen bestod av seks deltakere, og hadde følgende sammensetning:

- Tre brukere fra SKD som ikke hadde brukt Altinn før:  
To kvinner med 8 og 11 års fartstid i SKD, en mann med 3 års ansiennitet i SKD. Alle bruker datamaskiner til daglig og kan en del om skatt.
- Tre eksterne brukere som ikke hadde brukt Altinn før:  
To kvinnelige studenter i internasjonal markedsføring (24 og 25 år) som bruker datamaskiner og internett til daglig. En mannlig gründer (27 år), utdannet siviløkonom og master i IT, jobber med utvikling av programvare.

Evalueringen var lagt opp i to deler. I første del presenterte vi prinsipper og prototyper, hvorpå deltakerne ble bedt om å svare på en spørreundersøkelse individuelt. Testlederne og observatørene gikk så gjennom svarene før del to av sesjonen.

Andre del var lagt opp som en gruppeoppgave/fokusgruppe. Vi presenterte svarene fra første del og bad om innspill og kommentarer til de forskjellige alternativene som var gjennomgått. Hvert punkt ble så diskutert i plenum. Hensikten med dette var å få frem uklarheter og hva begrunnelsen bak de individuelle svarene i spørreundersøkelsen var.

### **Oppsummering av arbeidet i eSkjema SKD**

På bakgrunn av metodene over satte prosjektgruppa sammen et sett med retningslinjer [Skattedirektoratet 2005b]. Disse retningslinjene er *"en selvstendig videreføring og intern tilpassning av de såkalte ELMER-prinsippene [...]"* [ibid: 4].

### **Analyse av skjema (innholdsanalyse)**

Innholdsanalyse [Rosenfeld og Morville 2002: 221] er en metode der vi ser på nettopp *innholdet* i det vi skal strukturere. Innholdsanalyse er en bottom-up tilnærming.

Jeg har sett på selvangivelsen for næringsdrivende 2004. Jeg har tatt for meg både det elektroniske skjemaet i Altinn og papirutgaven av skjemaet.

## Heuristisk evaluering

Det er blitt foretatt en heuristisk evaluering av skjemaautformingen. Denne ble gjort samtidig med evalueringen av webportalen. Jeg refererer derfor til metodebeskrivelsen i kapittel 5 for informasjon om metoden. Resultater som berører evalueringen av skjemaautformingen presenteres senere i dette kapitlet.

## 6.4. Resultater

### Erfaringer med Altinn

Resultatene i erfaringsrapporten [Udjus 2005] var av svært forskjellig karakter. Med det mener jeg at noen av kommentarene var spesifikke, mens andre var generelle. Her følger en kort oppsummering av problemområder som ble funnet.

#### Konsistens

Et av hovedproblemene var *konsistens* i skjemaene. Med Skatteetatens store antall skjema i Altinn (over 80 stykker på daværende tidspunkt) og ingen overordnet prosess for konsistens i skjemaene var det til dels store forskjeller i hvordan skjemaene var utformet. Inkonsistens fantes på alle plan: i bruk av forkortelser, gruppering av elementer, nummerering av sider og poster / felter, og i overføringer mellom felter.

#### Struktur

Et annet stort problem var strukturen på skjemaene. Dette gjaldt på et overordnet nivå der noen skjema hadde for høyt antall sider, mens andre hadde for lavt antall sider (og dermed for mye informasjon på hver side). Videre var det enkelte skjema som hadde ulogisk rekkefølge på sidene.

Det var også problemer med strukturen på lavere nivå, altså innen en enkelt side. Enkelte steder var grupper som burde utgjøre én side delt over flere sider, mens i andre tilfeller var for mye informasjon dyttet inn i en side. De samme problemene viste seg i hvilke elementer som utgjør én logisk gruppe innen en side.

#### Layout

Den siste hovedgruppen med problemer var bruken av visuelle virkemidler. Deler av disse problemene skyldtes problemer med *konsistens* og *struktur*, og berørte således også de foregående punktene.

Eksempler på problemer her var for eksempel problemer med sideveis scrolling på grunn av brede tabeller. Dette skyldtes som regel at skjemaene i Altinn har beholdt tabellayouten til papirutgaven av samme skjema.

Videre var visuell layout av grupper gjort ved bruk av forskjellige virkemidler. Enkelte ganger var ble det brukt rammer/bokser, mens det i andre tilfeller ble benyttet luft.

### Heuristisk evaluering

Funn fra den heuristiske evalueringen finnes i kapittel 5.3 på side 57. Der finnes funn som dekker både webportalen og utformingen av skjema. Funn som gjelder skjemautforming summeres opp og diskuteres lenger ned i diskusjonsdelen av inneværende kapittel.

### Analyse av skjema

Som vi så av erfaringene med Altinn sirkulerte problemene rundt tre hovedområder: konsistens, struktur og layout. Også selvangivelsen for næringsdrivende har en del konsistensproblemer med andre skjema. Men fordi vi kun ser på ett skjema her vil jeg ikke berøre konsistensproblematikk i særlig grad i denne analysen.

De største problemene går på struktur i skjemaet og hvordan visuelle virkemidler er blitt brukt til layout av skjemaet.

Under analysen av skjemaet har jeg sett på både papirutgaven og skjemaet i Altinn.

### Struktur

#### Overordnet oppbygning av skjemaet

Skjemaet er delt opp i 21 sider, med følgende struktur (legg merke til at det er enkelte lenker som står oppført flere ganger. Dette er ikke feil.):

- Generell informasjon
- 1.4 Barn som du forsørger
- 1.5 Andre opplysninger
- 1.6 Personinntekt - Foretak
- 2.1 Lønn, naturalytelser mv.
- 2.2 Egne pensjoner, livrenter i arbeidsforhold mv.
- 2.4 Barns lønnsinntekt
- 2.6 Mottatte bidrag, livrenter utenfor arbeidsforhold, barnepensjon mv.
- 2.7 Næringsinntekter, Sykepenges mv.
- 2.8 Inntekter av bolig og annen fast eiendom
- 3.1 Kapitalinntekter og andre inntekter
- 3.2 Fradrag i tilknytning til arbeidsinntekt mv. og underskudd i næring. Minstefradrag
- 3.3 Kapitalkostnader og andre fradrag
- 3.3 Kapitalkostnader og andre fradrag
- 3.4, 3.5 og 3.6 Alminnelig inntekt, særfradrag og grunnlag for beregning av kommune-, fylkes- og fellesskatt
- 4.1 Bankinnskudd, kontanter, verdipapirer mv.

- 4.2 Innbo og løsøre
- 4.3 Faste eiendommer, anlegg, skog mv.
- 4.3 Faste eiendommer, anlegg, skog mv.
- 4.4-4.7 Formue – Driftsløsøre, annen formue, skattepliktig formue i utland og nettoformue.
- 4.8 Gjeld

## 6.5. Diskusjon

### Analyse av skjema

#### Struktur

Det er verdt å merke seg at forslaget som jeg kommer frem til her *fortsatt må jobbes videre med*. Dette burde gjøres sammen med jurister og personell fra fagavdelingene for å sikre at skjemaet forholder seg til regelverket og er korrekt fra etatens side.

Jeg har valgt å *kun* se på den overordnede strukturen i skjemaet i denne oppgaven. Jeg har ikke tatt for meg strukturen innen en side i skjemaet siden dette raskt ville blitt en svært stor oppgave i et skjema som selvangivelsen. I stedet har jeg valgt å fokusere på den overordnede strukturen i skjemaet og presentasjonen av denne strukturen i venstremenyen.

#### Overordnet oppbygging av skjemaet

For brukeren kan sideinndelingen virke merkelig, spesielt siden det noen sider som har samme overskrift. I tillegg er det vanskelig for brukeren å få oversikt over skjemaet fordi overskriftene kuttes (se Figur 34).

Rekkefølgen på postene følger papirutgaven. Papirutgaven har forøvrig fire sider, og er inndelt i fem overordnede temaer: generell informasjon, personinntekt, alminnelig inntekt, fradrag, formue og gjeld.

Det er flere problemer med dagens struktur på skjemaet:

- Skjemaet utnytter ikke sporvalg
- Det er flere lenker med samme navn i menyen
- Det er vanskelig å få oversikt over skjemaet



**Figur 34: Venstremenyen i selvangivelsen for næringsdrivende 2004**

I Figur 34 ser vi at venstremenyen gir dårlig oversikt over innholdet i skjemaet. Problemet er at titlene er lange og at de starter med nummer. Det verste tilfellet er punktet som heter "3.4, 3.5 og 3.6 A..." Det er en intetsigende tittel som ikke hjelper brukeren. Vi har berørt de tekniske aspektene ved menyen i kapittel 5. Her skal vi konsentrere oss om å lage gode navn på sidene, og lage en struktur på skjemasidene som er fornuftig.

Det er ikke enkelt å lage en god struktur på skjemaet som fungerer godt på nettet. Det er spesielt to problemer som melder seg når vi prøver å restrukturere skjemaet:

- 1) Vi kan ikke ha for mange sider på grunn av menyen
- 2) Enkelte av punktene inneholder lite informasjon. Det er dumt å ha sider med lite informasjon. Derfor vil enkelte sider smelte sammen.

Punkt 1 kan være problematisk å komme forbi i et så stort skjema som selvangivelsen er. Punkt 2 skaper problemer med nummereringen i menyen.

For å få et løsningsforslag har jeg gjort følgende:

- Smelte sammen duplikatsider

- Gruppere etter tema fra papirutgaven
- Lete etter sider som kan være kandidater for sporvalg
- Omstrukturere skjemaet der det virker hensiktsmessig

Punkter som står mellom parenteser er punkter som det blir spurt om på sporvalgssiden. Disse sidene vil ikke komme opp dersom brukeren ikke har behov for dem. Det er også enkelte punkter som har en skråstrek mellom seg (/). Dette er punkter jeg mener kan komme på samme side.

Før brukeren kommer inn i selve skjemaet kommer en egen side med spørsmål. Svarene på disse spørsmålene avgjør hvilke sider som skjemaet settes sammen av (sporvalg).

#### Sporvalg

- **Generell informasjon**
  - Personalia
  - (Ektefelle)/(samboer)
  - (Barn som du forsørger)
- **Personinntekt**
  - Personinntekt fra foretak
  - Lønn
  - (Barns lønnsinntekt)
  - Egne pensjoner
- **Alminnelig inntekt**
  - Mottatte bidrag
  - Næringsinntekter / Sykepenger
  - Inntekt av bolig og fast eiendom
  - Kapitalinntekter og andre inntekter
- **Fradrag**
  - Fradrag i arbeidsinntekt og minstefradrag
  - Underskudd i næring
  - Kapitalkostnader og andre fradrag / (Særfradrag)
- **Formue og gjeld**
  - Bankinnskudd, kontanter og verdipapirer
  - Innbo og løsøre
  - Faste eiendommer, anlegg, skog mv.
  - (Driftsløsøre og eiendeler i næring)
  - Annen formue / (Formue i utlandet)
  - Gjeld
- **Andre opplysninger**
- **Oppsummering**

3.4, 3.5, 3.6 droppes og legges inn som mellomsummer, siden denne siden er en opplisting av beregnede tall.

Jeg har flyttet "Andre opplysninger" til slutt i skjemaet siden det føles mer naturlig å ta dette til slutt. I "Oppsummering" kommer en oppsummering av nøkkeltall fra hele skjemaet.



I Figur 35 har laget en skisse for hvordan det ovennevnte løsningsforslaget kan se ut etter en innledende sporvalgsdialog. Løsningsforslaget baserer seg på en del av forslagene jeg kom til webportalen i kapittel 5. Jeg har blant annet fjernet fanene for skjema og vedlegg og fjernet vedleggstabellene. Videre har jeg også fjernet indikatorpila som lå helt til venstre i menyen. Jeg har heller ikke kutting av punktene i menyen, slik at teksten i menyen står fullt ut.

Grepene jeg har gjort med menyen har frigjort verdifull plass som benyttes av punktene i menyen.

Jeg har også valgt å fjerne nummereringen av punktene i menyen. Dette vet jeg er et omdiskutert valg fra fagavdelingenes side. Det refereres ofte til postnummer, derfor kan det være nyttig å ha nummer i menyen. Jeg har likevel valgt å fjerne de. Den viktigste grunnen var for oversikten i menyen – mer innhold er lik mer støy. Tanken var å få menyen så oversiktlig som mulig. I tillegg var nummereringen et problem når jeg nå har stokket om på rekkefølgen av skjemasidene og bruker sporvalg. Hvordan skal etatene kunne bruke et fornuftig nummersystem da? Dette er noe som må ses nærmere på av andre.



Figur 35: Løsningsforslag for ny struktur og navngiving av skjemasider i selvangivelsen for næringsdrivende

## Heuristisk evaluering

### Oppsummerte funn

Dette er oppsummerte funn fra den heuristiske evalueringen. Funnene som listes opp her tar for seg innholdet i skjemaet. Det er kun tatt med funn som har alvorlighetsgrad 3 (stort problem) eller 4 (brukervennlighetskatastrofe).

### Avklaringsspørsmål/sporvalg

21. Hvorfor må alle brukere innom "barns lønnsinntekt" og "mottatte livrenter"?  
Brukere må ta stilling til ting som mottaker enten vet fra før (og da kan stille spørsmål om) eller som bruker kjøpt kunne svart ja/nei på i begynnelsen av skjemaet.
- Bryter med: *tilpassning (ny)*
  - Alvorlighetsgrad: 3

### Kalkuleringer

22. Det er lite konsistens i hva som kalkuleres og hva som ikke kalkuleres.
- Bryter med: 1) *konsistens*
  - Alvorlighetsgrad: 3
23. Når man fyller ut skjemaet i tilfeldig rekkefølge er det av og til vanskelig å skjønne at enkelte felter henter data fra tidligere felter i skjemaet.
- Bryter med: 9) *status*
  - Alvorlighetsgrad: 3

### Hjelpetekst

24. Mye av hjelpteksten gir for lite hjelp i forhold til hva brukeren lurte på i utfyllingsøyeblikket.
- Bryter med: 12) *hjelp*
  - Alvorlighetsgrad: 3

## Analyse og løsningsforslag

### Avklaringsspørsmål/sporvalg

21. Hvorfor må alle brukere innom "barns lønnsinntekt" og "mottatte livrenter"?  
Brukere må ta stilling til ting som mottaker enten vet fra før (og da kan stille spørsmål om) eller som bruker kjøpt kunne svart ja/nei på i begynnelsen av skjemaet.
- Bryter med: *tilpassning (ny)*
  - Alvorlighetsgrad: 3

Dette kunne vært løst ved at det legges inn avklaringsspørsmål i begynnelsen av skjemaet. Det er dette ELMER-rapportene refererer til som "sporvalg" [Nygaard 2001b: 9]. Hovedformålet med slike avklaringsspørsmål er å fjerne unødvendige spørsmål, og dermed gjøre utfylling enklere og kjappere for utfylleren. Det er flere spørsmål i selvangivelsen for næringsdrivende som kunne blitt luket ut på denne måten.

Ved bruk av sporvalg må vi være forsiktige slik at vi ikke ender opp med at ett sporvalgsspørsmål erstatter ett vanlig spørsmål. Da vil vi ikke ende opp med noen besparelse for brukeren. Hvorvidt vi skal introdusere sporvalgsspørsmål blir derfor alltid en avveining om det innfører en reell besparelse for brukeren. Det er heller ikke hensiktsmessig med for mange sporvalgsspørsmål i begynnelsen av skjemaet.

### Kalkuleringer

Kalkuleringer eller beregninger er uttrykk som brukes når skjemaet har innebyggede funksjoner for å regne ut tall for brukeren. Dette kan være enkle summeringsfunksjoner som legger sammen tall, men det kan også være mer komplekse regnestykker, som beregner forskjellige fradrag m.m.

22. Det er lite konsistens i hva som kalkuleres og hva som ikke kalkuleres.
- Bryter med: 1) *konsistens*
  - Alvorlighetsgrad: 3

**2.8 Inntekt av bolig og annen fast eiendom**

2.8.1 Inntekt av prosentliknet boligeiendom, herunder egen bolig i borettslag, boligaksjeselskap eller boligsamele

Kommune	Likningsverdi opp til 451 000	Likningsverdi opp til 451 000, - 90 000 x 2,5%	Beløp
OSLO	451000		
	Likningsverdi som overstiger 451		
	Likningsverdi som overstiger		
			2.8.1 Skattepliktig beløp

Hvis du vil registrere flere, velg rett tabell fra menyen Tabeller som du finner til venstre.

Figur 36: Eksempel på at det ikke er lagt inn kalkuleringer over alt.

Skatteetaten har for det meste vært flinke til å legge inn kalkuleringer i skjemaene sine. Likevel dukker det opp steder der kalkuleringer ikke finnes. Dette irriterer brukerne når det likevel er så mange andre steder systemet beregner tall for dem. Det bør legges inn kalkuleringer der det kan legges inn.

23. Når man fyller ut skjemaet i tilfeldig rekkefølge er det av og til vanskelig å skjønne at enkelte felter henter data fra tidligere felter i skjemaet.
- Bryter med: 9) *status*
  - Alvorlighetsgrad: 3

I Altinn kan etaten velge om de vil at skjemaene skal fylles ut i en bestemt rekkefølge eller om de kan fylles ut i valgfri rekkefølge. I [Nygaard 2001b: 13] anbefaler ELMER-gruppa at skjema bør ha fri utfylling. De fleste skjemaene til Skatteetaten er åpne og kan fylles ut i den rekkefølgen brukeren selv ønsker. Det at brukeren selv kan velge hvilken rekkefølge han vil fylle ut i gir frihet, men øker også ansvaret til etatene og designerne. Fri navigering kommer med større muligheter for misforståelse fordi

brukerne kan følge andre løyper enn designerne har tenkt. Derfor blir det enda viktigere å klargjøre *hva* som skal skje på en gitt side, og hvor data hentes fra.

Problemet som ble fastslått over kan hende dersom etaten ikke har gjort det klart at et felt henter data fra andre steder i skjemaet. Det er viktig at det klart informeres når et felt henter data fra andre felter, dersom det ikke er opplagt ved at data kommer fra et nærliggende felt.

### Presentasjon av preutfylte felter

Vi har foreslått i eSkjema SKD å presentere deler av den preutfylte informasjonen som ren tekst, og ikke som utgråete felter. Informasjonen kan uansett ikke endres på grunn av regler for oppdatering av registre, og fordi dette ofte krever separate endringsmeldinger. Det er derfor liten grunn til å la disse feltene, som for eksempel navn og adresse til å fremstå som "halvt" endringsmulige (altså i endringsfelter som er låste). Begrunnelsen bak dette forslaget er at etaten har mottatt reaksjoner fra brukerne der de ønsker å endre disse låste feltene (for eksempel dersom informasjonen som står der ikke lenger stemmer).



Personalia ?	
1.1.1 Navn	NORDMANN OLA
Adresse	DRAMMENSVEIEN 1
Postnr.	0001
Poststed	OSLO
1.1.2 Boligadresse 1. november 2003	
Postadresse 1. november 2003	

**Figur 37: Slik ser preutfylte felter i dag.**

Ved å presentere slike felter som ren tekst signaliserer vi til brukerne at dette ikke kan endres her overhodet. Dessverre er det slik at skjemamotoren ikke klarer å preutfylle til vanlig tekst; det må være i et form-element. (dette kan forøvrig jukses til ved å endre utseendet på input-felter i CSS, men det vil være en hack. Det beste ville være om dette kunne legges inn som ren tekst (ikke minst med tanke på universell utforming).

Når teksten likevel ikke kan endres kan det også gjøres noe med oppsettet på feltene, slik at vi gjør det enda mer klart for brukeren at dette er data som ikke kan endres. I Figur 38 er det satt opp et forslag til hvordan slik informasjon kan se ut.

? Selskap		? Regnskapsfører	
<b>Org.nummer:</b>	436053054	<b>Navn:</b>	Ketil Storvik
<b>Navn:</b>	Bodø Marin	<b>Adresse:</b>	Rådyrveien 12
<b>Adresse:</b>	Bryggeveien 45	<b>Poststed:</b>	8433 Vesterøy
<b>Poststed:</b>	8000 Bodø		

Figur 38: Forslag til hvordan preutfylt informasjon som ikke kan endres kan se ut.

### Hjelpetekst

24. Mye av hjelpeteksten gir for lite hjelp i forhold til hva brukeren lurte på i utfyllingsøyeblikket.

- Bryter med: 12) *hjelp*
- Alvorlighetsgrad: 3

I følge ELMER skal hjelpeikonene og den tilhørende hjelpeteksten hjelpe utfylleren i utfyllingsøyeblikket – det er dette de kaller for "selvvalgt hjelpetekst". Selvvalgt hjelpetekst er i følge ELMER [Nygaard 2001b: 23] :

*"Tilleggsinformasjon til et spørsmål, som utfylleren selv velger å bla opp i en separat veiledning eller «klikke fram» i et elektronisk skjema."*

I ELMER-demoen var den meste av den selvvalgte hjelpeteksten skreddersydd for enkeltspørsmålene i skjemaet. Tanken var å ha spisset hjelp som går på en direkte problemstilling (det brukeren lurte på i utfyllingsøyeblikket). Brukeren skal få svar på det han lurte på så fort som mulig når han først ber om hjelp.

De fleste skjemaene til Skatteetaten har *ikke* egen hjelpetekst som er skrevet spesielt for web. For det meste dreier deg seg om en oppstykket papirveiledning som er puttet inn.

I Figur 39 kan spørsmål som utfylleren lurte på være "hva er livrenter?". Eller "hva defineres som et bidrag?", "hva er føderåd?".

Det kan diskuteres hvor langt etaten bør gå i å lære opp utfyllerne sine inne i skjemaene sine. Skatteregler har aldri vært enkelt. For *brukerne*, derimot, er det klart at mange av spørsmålene som dukker opp i skjemaene spør om ting de ikke kjenner til. Skjemaene retter seg tross alt mot store grupper i befolkningen. Selv med utstrakt bruk av sporvalg vil brukerne støte på emner de ikke kjenner til. Hjelpetekst kan potensielt avhjelpe dette problemet.

Figur 39: Hjelpetekst aktivert

Det vil være ressurskrevende for etatene å lage gode hjelpetekster. Dette er en av grunnene til at Skatteetaten ikke har egne hjelpetekster for web (enda). Dagens hjelpetekster har stort forbedringspotensiale i forhold til å svare på brukerens spørsmål i utfyllingsøyeblikket. Dagens hjelpetekster er tatt fra papirrettledningen, som er generell i natur.

Nielsen oppgir tre hovedretningslinjer for tekst på web [Nielsen 1999a: 101]:

- *“Be succinct. Write no more than 50 percent of the text you would have used to cover the same material in a print publication.*
- *Write for scannability. Don't require users to read long continuous blocks of text; instead, use short paragraphs, subheadings, and bulleted lists.*
- *Use hypertext to split up long information into multiple pages.”*

Deler av disse punktene kan ikke etatene gjøre noe med. Som vi så i kapittel 5 går det ikke an å formattere hjelpeteksten eller legge inn lenker. Dette er klare behov i hjelpesystemet som det er viktig å utbedre. Når det er gjort kan etatene ta fatt på arbeidet med å skrive om hjelpetekstene for web.

## 6.6. Oppsummering

I dette kapittelet har jeg sett på forbedringspotensialet i skjemaautforming i Altinn. Jeg har sett på ett spesifikt skjema som eksempel: selvangivelsen for næringsdrivende 2004.

Først viste jeg et forslag til hvordan selvangivelsen for næringsdrivende kunne bygges om slik at skjemastrukturen var mer oversiktlig ved hjelp av sporvalg og ny struktur. Ved å utnytte ekstra plass jeg frigjorde i menyen i kapittel 5 fikk jeg plass til

grupperingsoverskrifter. Både strukturen i skjemaet og oversikten i menyen ble på denne måten bedre.

Etter dette gikk jeg gjennom resultater fra den heuristiske evalueringen som omhandlet skjemaautforming:

ID	Problem	Løsningsforslag
21	Hvorfor må alle brukere innom sider med spørsmål de ikke trenger å bli spurt om?	Ved å legge inn avklaringsspørsmål (sporvalg) vil dette kunne løses.
22	Det er lite konsistens i hva som kalkuleres og hva som ikke kalkuleres.	Det bør legges inn kalkuleringer de stedene i skjemaet der det ikke er gjort det enda.
23	Når man fyller ut skjemaet i tilfeldig rekkefølge er det av og til vanskelig å skjønne at enkelte felter henter data fra tidligere felter i skjemaet.	I tilfeller der data hentes fra andre steder i skjemaet må det gjøres klart at data hentes fra andre steder. Dette kan gjøres ved å legge inn tekst ved det aktuelle feltet som forteller hvor det hentes data fra.
24	Mye av hjelpeteksten gir for lite hjelp i forhold til det brukeren lurte på i utfyllingsøyeblikket.	Hjelpetekstene må skrives om for web. Konkret betyr dette å gjøre tekstene kortere og mer spisset i forhold til det brukerne lurte på i utfyllingsøyeblikket. Tekstene må også splittes opp og benytte overskrifter, lister og formattering. Dette er ikke mulig i dagens tekniske løsning.



## 7. FORBEDRINGSPOTENSIALET I UNIVERSELL UTFORMING

*"The power of the Web is in its universality. Access by everyone regardless of disability is an essential aspect."*

– Tim Berners-Lee,  
Leder av W3C og oppfinneren av World Wide Web

### 7.1. Introduksjon

I dette kapitlet vil jeg se på Altinn og forbedringspotensialet i universell utforming. Jeg vil ta for meg hva universell utforming er, hvorfor det er viktig og litt om den politiske IT-strategien når det gjelder tilgjengelighet for alle. Jeg vil så gå gjennom metoden jeg har brukt til analyse av Altinn med hensyn på tilgjengelighet og presentere resultater og diskusjon rundt resultatene.

#### Hva er universell utforming og tilgjengelighet?

I denne oppgaven vil jeg bruke begrepene "universell utforming" og "tilgjengelighet" om hverandre. På engelsk brukes for det meste uttrykket "accessibility", den mest brukte norske oversettelsen er "tilgjengelighet".

"Accessibility" er av W3C definert som:

*"Web accessibility means access to the Web by everyone, regardless of disability".*

Videre defineres "universell tilgang" som:

*"Universal Access: To make the Web accessible to all by promoting technologies that take into account the vast differences in culture, languages, education, ability, material resources, access devices, and physical limitations of users on all continents."*

–[www.w3.org/Consortium/](http://www.w3.org/Consortium/)

Vi ser altså at "universell utforming" er et videre begrep enn "tilgjengelighet". I denne rapporten vil jeg først og fremst fokusere på tilrettelegging av nettsider slik at folk med andre fysiske behov kan få tilgang til Altinn.

I den offisielle norske språkbruken finner man ikke så mange referanser til tilgjengelighet. På norsk brukes uttrykket "universell utforming" eller "universell tilgang" i større grad. Jeg vil derfor bruke alle disse begrepene om hverandre i denne rapporten.

*Universell utforming/tilgjengelighet* er begreper som betyr å tilrettelegge programmer og nettsider slik at brukere med ulike former for funksjonshemming kan få tilgang til og benytte systemene.

Prinsippene som legges fram i ELMER og i SSBs retningslinjer legger lite eksplisitt vekt på universell utforming. ELMER og SSB har fokusert på mer generelle visuelt orienterte prinsipper. Noen av disse prinsippene hjelper også på universell utforming, men siden prinsippene for det meste er generelle er det også mye som forblir usagt. Universell utforming har, i likhet med prinsipper for gode brukergrensesnitt, både en generell og en konkret side. Den generelle siden tegner opp overordnede prinsipper som "konsistent navigasjon" og "lettfattelig tekst". Den konkrete siden omhandler hvordan nettsidene burde implementeres og legger opp anbefalinger for hvordan koding burde gjøres osv.

World Wide Web Consortium (W3C), de som har det overordnede ansvaret for de teknologiske standardene som internett bygger på har et program som heter Web Accessibility Initiative (WAI). WAI kommer med en rekke anbefalinger for hvordan nettet kan bedre støtte funksjonshemmede, svaksynte og eldre mennesker.

### **Hvorfor er universell utforming viktig for offentlige nettsider?**

Universell utforming er viktig av flere grunner. Først og fremst gjelder det å gi flest mulig mennesker muligheten til å bruke nettet.

Tilgjengelighet for alle og universell utforming er forankret i eNorge 2005. Rapporten sier blant annet: "Alle skal kunne utnytte informasjonsteknologiens muligheter". Videre sies det at "Det må i større grad fokuseres på brukernes – kundenes – behov i utviklingen..." [NHD 2002: 6].

Dette sier ikke bare at vi må lage brukersentrerte tjenester, det sier også at vi må ta hensyn, slik at *alle* kan benytte seg av de nettbaserte tjenestene som tilbys. Likevel fastslås det enda klarere at Staten har et ansvar for å tilrettelegge tjenestene sine: "Offentlige internettsider skal være brukervennlige og oppfylle internasjonale retningslinjer for design og universell utforming. Dette vil også ivareta funksjonshemmedes behov bedre." [NHD 2002: 24]. De internasjonale retningslinjene det her pekes til er WAIs retningslinjer.

USA endret i 1998 deler av en lov som gjorde at alle offentlig tilgjengelige nettstedet må være tilgjengelige for alle – også de med handikap. Dette er kjent som "Section 508" [USC 1998].

I England vedtok de Disability Discrimination Act i 1995. Denne begynte å gjelde engelske nettsteder fra oktober 2004 [DWP 1995].

I tillegg til det etiske ansvaret staten har, er det altså klart at vi har en klar politisk forankring for å gjøre nettjenestene til staten tilgjengelig til så stor gruppe som mulig. Dette innebærer konkret at vi må ta hensyn til universell utforming under utvikling.

### **Praktiske og økonomiske fordeler ved universell utforming**

Selv om det er etisk riktig, i tillegg til å være politisk forankret er det også en rekke fordeler ved å ta hensyn til universell utforming. Jeg tar kort for meg disse punktene under.

#### **1) Lettere å vedlikeholde**

Ved korrekt bruk av webstandarder vil man få en løsning som har god separasjon av innhold og presentasjonsinformasjon. Dette tilsvarer en god systemarkitektur som gjør at vedlikehold av systemet blir enklere. Dette igjen gjør at vedlikeholdskostnader kan senkes.

#### **2) Fremtidsrettet**

Korrekt bruk av webstandarder gjør også løsningen mer fremtidsrettet. Ved å følge disse standardene sørger man ikke bare for at løsningen fungerer optimalt i dag, man sørger for å maksimere mulighetene for at løsningen vil fungere med neste generasjons enheter. Håndholdte enheter som mobiltelefoner og PDAer spiller en stadig større rolle på nettet. Måten vi bruker nettet på endrer seg; trenden går fra stasjonær til mobil bruk.

#### **3) Raskere nedlasting**

Et utfall av "korrekt" koding er at filstørrelsen til nettsidene vil kunne krympe. Lavere filstørrelse resulterer i flere gevinster:

- Bedre bruksopplevelse  
Trege nettsider plager fortsatt brukere. Selv de med bredbånd vil sette pris på enda raskere nettsider.
- Tar hensyn til mobile brukere  
Håndholdte enheter har som regel mindre kapasitet enn stasjonære maskiner.
- Lavere kostnader  
Mindre filstørrelser, betyr mindre data overført. Dette betyr lavere kostnader i form av lavere linjeleie.

Her finnes det mye empiri. Microsoft redesignet for ikke så lenge siden nettsidene deres ([www.microsoft.com](http://www.microsoft.com)) og implementerte webstandarder. Det er mange eksempler på hva bedre koding kan få til. Et eksempel som tar for seg fordelene er "Throwing Tables Out The Window" [Bowman 2004].

#### **4) Brukervennlighet og tilgjengelighet overlapper**

Å fokusere på universell utforming er ofte det samme som å fokusere på brukervennlighet. For webutvikling er det slik at mange av teknikkene for å lage mer brukervennlige sider overlapper med det å lage mer tilgjengelige sider. Ved å fokusere på universell utforming, vil man derfor kunne gjøre sidene mer brukervennlige samtidig [Moss 2004].

### 5) God PR

Universell utforming er fortsatt nytt.. Det er fortsatt slik at det er unormalt at nettstedet tar hensyn til disse gruppene. Å ivareta interessene til brukerne vil derfor skape god PR og positiv omtale rundt produktene.

### Hvem blir berørt av universell utforming?

Universell utforming berører ikke bare funksjonshemmede. Det påvirker langt flere enn de som tradisjonelt faller under samlebegrepet funksjonshemmede.

Over 10% av nettbrukerne i verden kan sies å ha en funksjonshemning [iCan 2004].

Fra siste levekårsundersøkelse i Norge [SSB 2002]:

- 3% har nedsatt syn
- 3% har nedsatt hørsel
- 10% har nedsatt bevegelsesevne
- 3% har store konsentrasjons- eller hukommelsesproblemer

I Storbritannia gjelder følgende tall [Watchfire 2004]:

- 14% regnes som funksjonshemmede
- 9% er fargeblinde
- 21% er over 60 år
- 4% har problemer med synet

Det er dog verdt å merke seg at det sannsynligvis er en del overlapp mellom disse gruppene.

De samme tallene fra W3C sier at andelen av nettbrukere med behov for universell utforming er mer enn 10 prosent (W3C/WAI 2003: slide 7). Dette er en stor gruppe. Det som er viktig å innse er at design for universell utforming er utforming for alle – også for oss selv. Mange av oss som ikke har funksjonshemninger i dag vil også havne i denne gruppa. Det å ta hensyn til universell utforming er med andre ord viktig for at vi som er unge i dag kan få glede av nettet når vi blir eldre.

## Altinn-kontrakten

I kravspesifikasjonen til Altinn, side 40 [Skattedirektoratet 2002] står det klart at Altinn skal ta hensyn til universell utforming og tilgjengelighet:

### *"Teknisk tilgjengelighet*

*Web-løsningen skal kunne benyttes med ulike typer nettlesere. Innholdet på Web-løsningen skal kunne leses selv om man skrur av funksjonalitet for å vise bilder. Ved bruk av for eksempel HTML eller XML skal Web-tjenesten følge W3C's gjeldende standarder. [...] Avvik skal kommenteres.*

*Web-løsningen skal følge W3C's gjeldende standard for tilgjengelighet og være merket i henhold til dette. [...] (Standarden sikrer bl.a. tilgang for synshemmede.)"*

Disse kriteriene er tatt fra "Kvalitetskriterier for offentlige webtjenester" [Statskonsult 2004] som er utarbeidet av Statens Informasjonstjeneste.

## Stortingsmelding 40

I stortingsmeldingen "Nedbygging av funksjons-hemmende barrierer" [Sosialdepartementet 2002-2003] foreslo regjeringen bl.a. en større satsing på å tilrettelegge IKT for personer med nedsatt funksjonsevne.

Blant virkemidlene foreslår regjeringen tilrettelegging av informasjon på Internett, samt at alle offentlige nettsider skal oppfylle retningslinjene fra Web Accessibility Initiative (WAI).

## W3Cs retningslinjer

World Wide Web Consortium (W3C) er organisasjonen som står bak mange av de standardene som internett bygger på. Blant disse standardene er språket som brukes for å bygge nettsider: HTML.

W3C har tre overliggende mål [W3C 1999-2004]:

1. *Universal Access: To make the Web accessible to all by promoting technologies that take into account the vast differences in culture, languages, education, ability, material resources, access devices, and physical limitations of users on all continents;*
2. *Semantic Web: To develop a software environment that permits each user to make the best use of the resources available on the Web;*
3. *Web of Trust: To guide the Web's development with careful consideration for the novel legal, commercial, and social issues raised by this technology.*

### **Web Accessibility Initiative (WAI)**

WAI er et program under W3C som arbeider for å bedre brukskvaliteten for mennesker med funksjonshemninger. De jobber med andre ord direkte med ett av de overliggende målene til W3C.

Dette programmet samarbeider med flere organisasjoner på et globalt nivå for å få maksimal gjennomslagskraft. WAI jobber med fem hovedområder: teknologi, retningslinjer, verktøy, utdanning og opplysning, og forskning og utvikling.

### **Web Content Accessibility Guidelines (WCAG) 1.0**

WAI utarbeidet og publiserte i 1999 et sett med retningslinjer for hvordan nettinhold bør utformes og implementeres for at det skal være mest mulig tilgjengelig. Disse retningslinjene har alle en prioriteringsmerking avhengig av hvor stor innvirkning de har på tilgjengelighet.

#### **Definisjon av prioritet 1**

Punkter med prioritet 1 er punkter som *må* tilfredsstilles. Hvis ikke vil en eller flere grupper umulig kunne få tilgang til informasjonen i dokumentet. Tilfredsstillelse av disse punktene er et krav for at enkelte grupper skal kunne bruke webdokumenter i det hele tatt.

#### **Definisjon av prioritet 2**

Punkter med prioritet 2 er punkter som *bør* tilfredsstilles. Ellers vil en eller flere grupper ha problemer med å få tilgang til informasjonen i dokumentet. Å tilfredsstille dette sjekkpunktet vil fjerne store hindringer for å få tilgang til webdokumenter.

#### **Definisjon av prioritet 3**

Punkter med prioritet 3 er punkter som *kan* tilfredsstilles. Hvis disse punktene ikke tilfredsstilles vil enkelte grupper ha mindre problemer med å få tilgang til informasjonen i dokumentet. Å tilfredsstille disse punktene vil forbedre tilgangen til webdokumenter.

### **Konformitetsnivå**

Konformitetsnivå spiller på hvor god støtte for universell tilgang en nettside har. Dersom prioritet 1 er tilfredsstilt kan nettstedet få en "A". Dersom nivå 2 er tilfredsstilt: "AA" (dobbel-A), og dersom alle nivåer er tilfredsstilt kan nettstedet få den beste karakteren: "AAA" (trippel-A).

## 7.2. Metode

Jeg har gjort en "preliminary review" av universell utforming i Altinn-løsningen. Jeg har da brukt fremgangsmåten spesifisert av W3C/WAI [W3C/WAI 1999] og brukt WAIs WCAG 1.0 som retningslinjer.

En såkalt preliminær gjennomgang er en test type som egner seg for rask kartlegging av mulige hindringer for universell tilgang på et nettsted. Testen er generell i natur, og er ikke ment som en omfattende test. Dersom man vil gjøre en omfattende test, burde det gjøres en såkalt "conformance evaluation", dvs. en test som i dybde sjekker nettstedet opp mot WCAG-retningslinjer og som også sjekker all kode m.m.

Jeg har gått litt dypere til verks enkelte steder i denne gjennomgangen. Spesielt under den manuelle testingen gjorde jeg en del inspeksjoner av koden for å finne ut hvorfor en del av feilene oppstod. Oppsummert kan jeg si at jeg har gått grundigere til verks enn en preliminær gjennomgang krever, men jeg har likevel ikke gått gjennom løsningen så nøye som en full "conformance evaluation" krever.

Min gjennomgang har vært bygget opp slik:

- 1) Velg ut et representativt utvalg av nettsider
- 2) Gjennomgang med en eller flere grafiske nettlesere
- 3) Gjennomgang med en eller flere talende nettlesere og tekstlige nettlesere
- 4) Test de utvalgte nettsidene i et automatisert verktøy som sjekker universell utforming
- 5) Manuell sjekk av nettsidene etter WCAG sjekkpunkter
- 6) Analyse

### Utvalget

Jeg har testet online-versjonen av Altinn (altså den som er tilgjengelig for publikum). Testingen har foregått i tidsrommet oktober – november 2004.

Fordi Altinn er en dynamisk løsning, som også krever at du logger inn fungerer automatisert testing av nettsidene dårlig. Jeg valgte derfor ut et utvalg av nettsider som jeg lagret kildekoden til. Denne kildekoden ble så testet i de automatiserte verktøyene. De samme sidene ble testet inne i selve løsningen i de manuelle testene.

Utvalget er basert på skjønn. Kriteriene for utvelgelsen har vært basert på innhold og funksjonalitet. Jeg har prøvd å få med meg alle typer sider som finnes i løsningen; både skjemasider, hjelpesider og også sider som jeg anser som sentrale: innlogging, hovedside og kontroll-sider.

Disse sidene var med i utvalget:

- Startsidene (den første siden du ser når du kommer inn på altinn.no)
- Logg inn (jeg fulgte linken "har vært her før")
- Legg til skjema (siden der du velger et skjema for utfylling som da flyttes til hovedsiden)

- Hovedsiden (siden der alle dine "aktive" skjema vises)
- Min profil
- Skjema: personlig selvangivelse for selvstendig næringsdrivende 2003
  - Generell informasjon ("startsidene" i skjemaet)
  - Gjeld (pkt: 4.8)
  - Personinntekt (pkt: 1.6)
  - Hjelp
  - Om skjema
  - Skjemakontroll

## Grafiske nettlesere

Den første delen av testen ble gjort i grafiske nettlesere. I denne delen sjekkes mulighetene for å bruke løsningen uten bruk av grafikk og lyd, muligheten for å øke og minke skriftstørrelse, oppløsning og vindusstørrelse. Her sjekkes også om løsningen tilfredsstiller krav til kontrast og bruk av farge. Til slutt testet jeg løsningen for hvor god den er til å bruke uten mus; altså kun med tastatur.

Nettlesere brukt i denne testen:

- Internet Explorer 6 (Windows)
- Opera 7.54 (Mac)

## Ikke-grafiske nettlesere

I denne delen av testen brukes tekst-baserte (ikke-grafiske) nettlesere og nettlesere som leser opp tekst høyt (brukes som regel av blinde og svaksynte).

Nettlesere brukt:

- Lynx 2.8.3
- Opera 7.54 (som emulert tekst browser)
- IBM Homepage Reader

## Automatiserte verktøy

De utvalgte sidene ble kjørt gjennom Bobby ([bobby.watchfire.com](http://bobby.watchfire.com)). Bobby er desktop-basert programvare som kan kontrollere nettsider mot flere regelsett for universell utforming, deriblant WCAG og 508. Jeg brukte en online-variant som ikke er så omfattende som skrivebordsløsningen. Regelsettet jeg brukte var WCAG1.0.

## Manuell testing

Automatiserte verktøy klarer ikke å dekke alt, spesielt siden det er mye som krever skjønn fra evaluators side. Jeg gjorde derfor også en manuell gjennomgang av retningslinjene i WCAG1.0. Her brukte jeg en sjekkliste med punkter fra WCAG1.0.



## 7.3. Resultater

### Grafiske nettlesere

Den første testen innebærer å slå av bilder/ grafikk i nettleseren for å se om løsningen lar seg bruke på denne måten. Mange av bildene mangler alternativ tekst. I tillegg var det vanskelig å bruke menyen/ navigasjonen i skjemaet fordi teksten var hvit, mot hvit bakgrunn.

Neste test var å bruke nettleserens muligheter for å justere tekststørrelsen. Inne på en skjemaside i Internet Explorer blir ikke all teksten blåst opp. Det er kun teksten i navigeringsmenyen i skjemaer som blir større. På største innstilling går teksten inn i hverandre, slik at det blir vanskelig å lese. Teksten i selve skjemaet blir ikke blåst opp. Ellers ser det ut til at alle andre sider har tekst som blir skalert opp.

I Opera fungerer endring av tekststørrelse bra, fordi programmet skalerer opp både tekst og bilder uavhengig av hvordan sidene er kodet. Dette gjør at det alltid er mulig å blåse opp størrelsen på en god måte, der alt beholder sine proporsjoner.

Endring av skjermopløsning fungerer ikke spesielt bra i noen av nettleserne. Det ser ut til at løsningen kun fungerer bra i 1024x768. Applikasjonen utnytter ikke høyere skjermopløsning. Lavere skjermopløsning gjør at man får horisontalt scrollefelt.

Neste test var å bruke løsningen i gråtoner. Jeg endret skjerminnstillingene til kun å vise gråtoner og prøvde å bruke løsningen. Kontrasten er gjennomgående god. Noen av feil-sidene var ikke like enkle å bruke, men elementer var likevel merket, bortfallet av farge spilte derfor ikke en kritisk rolle.

Siste test var å sjekke hvor enkel løsningen var å bruke uten mus. I Internet Explorer fungerte dette for det meste greit. Explorer har kun enkel navigering med tastatur, noe som betyr at man må tidvis klikke seg forbi veldig mange linker for å komme til riktig sted. Det skal likevel bemerkes at utviklerne har gjort et smart valg ved å sette rekkefølgen på tastatur-navigeringen slik at man alltid starter inne i selve skjemaet. Dermed kan man fylle ut skjemaet sammenhengende. Opera har bedre støtte for navigering uten mus. Her gikk det for det meste greit.

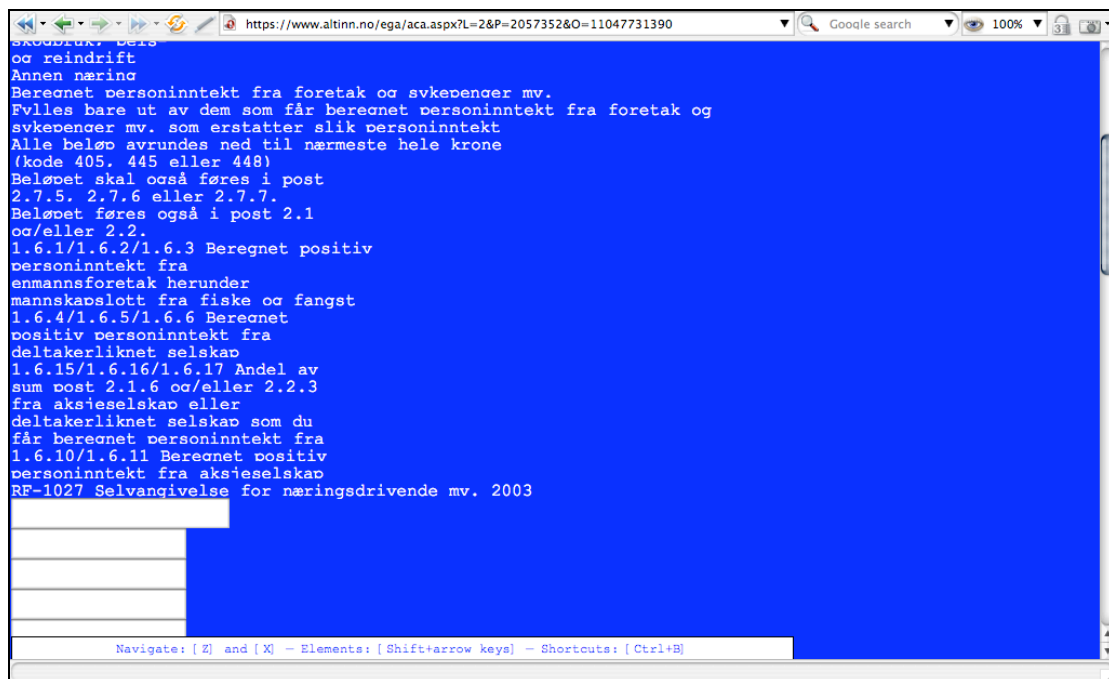
### Ikke-grafiske nettlesere

Den første nettleseren jeg prøvde å teste med var Lynx. Jeg kom inn på altinn.no, men da jeg skulle logge inn fikk jeg beskjed om at denne nettleseren ikke var støttet og at jeg måtte installere noe annet. Jeg fikk da kun valg mellom grafiske nettlesere (Internet Explorer, Opera, Netscape).

Jeg byttet derfor til Opera, som har mulighet for å emulere tekstleser. Jeg satte Opera fra "author mode" til "user mode" og slo på "emulate text browser", "accessibility layout" og "disable tables". Disse valgene fjernet all grafikk og fjernet alle kolonner, slik at teksten måtte leses serielt.

Testen går ut på å sjekke om all informasjonen som er tilgjengelig i grafikk-modus også er tilgjengelig i tekst-modus. I tillegg skal det sjekkes om informasjonen er presentert på en måte som gjør det mulig å lese det serielt. Resultatet kan ses i Figur 40.

Tekstfeltene jeg skulle fylle ut med informasjon er ikke kodet slik at de står ved siden av feltene der man fyller ut informasjonen. Det er dermed svært vanskelig å vite hvilke felter man skal fylle ut. I tillegg hadde jeg problemer med å trykke på enkelte knapper i denne modusen.



**Figur 40: Skjermdump av Altinn i tekstmodus (Opera). Legg merke til at ledetekstene ikke står i sammenheng med input-feltene.**

Jeg testet også Altinn med IBMs Home Page Reader. Dette er en høytlesende nettleser som bruker Internet Explorer som motor. Jeg hadde derfor ingen problemer med å komme inn på Altinn med denne nettleseren. Home Page Reader kan settes opp på mange måter. Den kan vise sidene med grafikk, slik siden ville blitt vist i en vanlig versjon av Explorer. Home Page Reader kan også vise sidene i tekst-format, slik som skjermbildet til Opera viste. Navigering kan foregå i begge modus, og med både tastatur og mus.

Hver gang jeg klikket meg videre fra en side fikk jeg opp en Javascript-feil, men sidene fungerte likevel.

### **Automatiserte verktøy**

Jeg har kun tatt for meg det som Bobby oppdaget som rene feil. Bobby markerer også såkalte "sjekkpunkter" – dette er punkter som ikke går an å kontrollere maskinelt.

Jeg har utelatt disse, fordi mange av disse punktene blir dekket av den manuelle testingen som kommer senere i rapporten.

Jeg testet 11 forskjellige sider (se utvalget på side 123), og har valgt å legge sammen antall feil for alle sider totalt, slik at vi får et visst overblikk over omfanget av feil innen de forskjellige kategoriene. Bobby deler opp feilene etter tre prioriteringer som svarer til WCAGs tre prioriteter (se side 122). Jeg tar for meg alle punktene tekstlig først, deretter kommer en oppsummering av resultatene i Tabell 1.

### **Prioritet 1**

Det er kun ett punkt under prioritet 1; Punkt 1.1 – "Provide alternative text for all images". Det betyr at all grafikk og alle bilder som brukes på nettsidene må ha en tekstlig beskrivelse. De som har behov for dette burde få en beskrivelse av hva de går glipp av dersom de ikke kan se grafikken, eller de velger å ikke laste den ned. Dette er noe som gjennomgående mangler.

### **Prioritet 2**

Prioritet 2 tar for seg flere områder. Punkt 1 (skriftstørrelser). Tekststørrelser og andre proporsjoner burde bruke relative størrelser for å fungere best mulig i forskjellige nettlesere og plattformer. Med 110 feil her, ser vi at det er utstrakt bruk av absolutte verdier. Resultatet er vanskeligheter med å øke tekst-størrelsen.

Punkt 2.2 og 2.4 tar for seg hvordan lenker presenteres. Det er viktig å presentere lenker på en fornuftig måte, slik at de kan leses utenfor kontekst. Dette hjelper både mennesker med- og uten handikap. Svaksynte/blinde vil ikke få noe særlig informasjon fra en link som sier "klikk her" – de blir nødt til å hoppe fram og tilbake i nettleseren for å vite hva den linken peker til. Mennesker uten synsproblemer vil også bruke lenger tid, fordi de fleste ikke leser all tekst – de "scanner" [Nielsen 1999a: 104, Krug 2000: 21].

Punkt 2.3 – "Make sure event handlers do not require use of a mouse" er viktig fordi det ikke er alle som bruker mus. Det er mange som bruker tastaturet for å navigere rundt på nettet. Dersom man bygger systemer som krever mus, kan man ekskludere store grupper. Dette gjelder ikke kun handikappede, det gjelder også folk som bruker andre enheter for å surfe; f.eks. mobiltelefoner.

Punkt 2.5 – "Provide an explicit label for each form control" forteller at det er viktig å merke alle elementer i et elektronisk skjema med en merkelapp som forteller hva du skal fylle ut. Hele 93 elementer fordelt på de 11 testede sidene mangler dette.

### **Prioritet 3**

Punkt 3.3 – "Provide a summary for tables" forutsetter at tabeller er brukt til det de egentlig er ment til; presentasjon av data i en matrise. I lang tid ble riktignok nettsider kodet slik at tabeller ble brukt til layout. Poenget med dette punktet er å summere opp datatabeller, slik at svaksynte/blinde kan velge å hoppe over en lang

tabell med data som ikke virker interessant; noe personer uten synsproblemer enkelt kan gjøre ved å bla videre. Dersom du bruker en høytlesende nettleser er ikke dette like enkelt.

Punkt 2-3 – "Identify the language of the text" kan virke trivielt, men er viktig for alle maskiner som skal lese teksten på nettsidene. Dette gjelder kanskje spesielt høytlesende nettleser som trenger å vite språket for å kunne lese opp teksten på en forståelig måte.

Punkt 3.3 – "Separate adjacent link with more than whitespace" er kun funnet ett sted i de testede sidene i Altinn. Den er viktig for at brukerne ikke skal gå glipp av linker.

Punkt 3.4 – "Include default, place-holding characters in edit boxes and text areas" er et interessant punkt. Dette punktet er (eller var) motivert med at nettleser har dårlig støtte for tomme skjema-elementer, derfor burde det brukes "fyllmasse". Dette skal hindre en del problemer som kan oppstå. Det å bruke slik "fyllmasse" strider med prinsippene fremlagt i blant annet ELMER. Jeg kommer tilbake til dette i diskusjonen, foreløpig konstaterer jeg bare at det ikke er vanlig med fyll-tekst i tekstfelter i Altinn.

<b>Priority 1 accessibility errors</b>	<b>Antall feil</b>
1. Provide alternative text for all images.	<b>51</b>
<b>Priority 2 accessibility errors</b>	
1. Use relative font sizing and positioning (% values) rather than absolute (pixels).	<b>110</b>
2. Create link phrases that make sense when read out of context.	<b>6</b>
3. Make sure event handlers do not require use of a mouse	<b>174</b>
4. Do not use the same link phrase more than once when the links point to different URLs.	<b>22</b>
5. Provide an explicit label for each form control.	<b>93</b>
<b>Priority 3 accessibility errors</b>	
1. Provide a summary for tables.	<b>103</b>
2. Identify the language of the text.	<b>11</b>
3. Separate adjacent link with more than whitespace.	<b>1</b>
4. Include default, place-holding characters in edit boxes and text areas.	<b>73</b>

**Tabell 1: Aggregererte testresultater fra Bobby**

### Manuell testing

Den manuelle gjennomgangen utfyller de automatiske testene som ble gjort gjennom Bobby. Også i den manuelle gjennomgangen brukte jeg WCAG1.0 som regelsett. Nok en gang er sjekkpunktene delt opp i tre prioriteter, der prioritet 1 er viktigst og

prioritet 3 er minst viktig. For hvert sjekkpunkt er det tre muligheter: hvis punktet er tilfredsstilt krysses det av for "yes", hvis ikke punktet er tilfredsstilt "no". "N/A" ("not applicable") brukes dersom punktet ikke er aktuelt for det testede nettstedet.

Denne testen er mer generell. Her står det en helhetsvurdering bak, i essens bedømmer jeg hele nettstedet og ikke enkeltsider i denne vurderingen. Mange av disse punktene er skjønnsmessige vurderinger, jeg vil derfor introdusere resultatene her og heller komme inn på eventuelle diskusjoner i neste kapittel.

Tabellene kan tolkes på følgende måte: For de punktene som er aktuelle kan tabellen tolkes slik at jo flere "ja" som er krysset av, jo bedre er løsningen rustet for å tilfredsstille kravene til universell utforming.

### Sjekkpunkter, prioritet 1

Dette er de punktene som *må* oppfylles for at nettstedet skal kunne tilfredsstille minimumskravene for at løsningen skal kunne brukes av alle og få en "A". For definisjon av prioriteringene, se side 122.

Det er seks punkter som er aktuelle for Altinn: 1.1, 2.1, 6.1, 14.1, 6.3 og 11.4. Av Tabell 2 kan vi se at av disse seks punktene, tilfredsstiller løsningen to punkter.

In general	Yes	No	N/A
1.1 Provide a text equivalent for every non-text element.		✓	
2.1 Ensure that all information conveyed with color is also available without color, for example from context or markup.	✓		
4.1 Clearly identify changes in the natural language of a document's text and any text equivalents (e.g., captions).			✓
6.1 Organize documents so they may be read without style sheets.		✓	
6.2 Ensure that equivalents for dynamic content are updated when the dynamic content changes.			✓
7.1 Until user agents allow users to control flickering, avoid causing the screen to flicker.			✓
14.1 Use the clearest and simplest language appropriate for a site's content.	✓		
<b>And if you use images and image maps (Priority 1)</b>	<b>Yes</b>	<b>No</b>	<b>N/A</b>
1.2 Provide redundant text links for each active region of a server-side image map.			✓
9.1 Provide client-side image maps instead of server-side image maps except where the regions cannot be defined with an available geometric shape.			✓
<b>And if you use tables (Priority 1)</b>	<b>Yes</b>	<b>No</b>	<b>N/A</b>
5.1 For data tables, identify row and column headers.			✓
5.2 For data tables that have two or more logical levels of row or column headers, use markup to associate data cells and header cells.			✓
<b>And if you use frames (Priority 1)</b>	<b>Yes</b>	<b>No</b>	<b>N/A</b>
12.1 Title each frame to facilitate frame identification and navigation.			✓
<b>And if you use applets and scripts (Priority 1)</b>	<b>Yes</b>	<b>No</b>	<b>N/A</b>
6.3 Ensure that pages are usable when scripts, applets, or other programmatic objects are turned off or not supported. If this is not		✓	

possible, provide equivalent information on an alternative accessible page.			
<b>And if you use multimedia (Priority 1)</b>	<b>Yes</b>	<b>No</b>	<b>N/A</b>
1.3 Until user agents can automatically read aloud the text equivalent of a visual track, provide an auditory description of the important information of the visual track of a multimedia presentation.			✓
1.4 For any time-based multimedia presentation (e.g., a movie or animation), synchronize equivalent alternatives (e.g., captions or auditory descriptions of the visual track) with the presentation.			✓
<b>And if all else fails (Priority 1)</b>	<b>Yes</b>	<b>No</b>	<b>N/A</b>
11.4 If, after best efforts, you cannot create an accessible page, provide a link to an alternative page that uses W3C technologies, is accessible, has equivalent information (or functionality), and is updated as often as the inaccessible (original) page.		✓	

Tabell 2: Testresultater fra manuell gjennomgang (prioritet 1)

**Sjekkpunkter, prioritet 2**

Prioritet 2 er punkter som *bør* tilfredsstilles for å få stempelet "AA". Resultatene fra den manuelle gjennomgangen er samlet i Tabell 3. Tabell 3 har totalt 30 punkter, hvorav 23 er aktuelle for Altinn. Av disse 23 kontrollpunktene tilfredsstiller Altinn kun 7 punkter under prioritet 2.

<b>In General (Priority 2)</b>	<b>Yes</b>	<b>No</b>	<b>N/A</b>
2.2 Ensure that foreground and background color combinations provide sufficient contrast when viewed by someone having color deficits or when viewed on a black and white screen. [Priority 2 for images, Priority 3 for text].	✓		
3.1 When an appropriate markup language exists, use markup rather than images to convey information.		✓	
3.2 Create documents that validate to published formal grammars.		✓	
3.3 Use style sheets to control layout and presentation.	✓		
3.4 Use relative rather than absolute units in markup language attribute values and style sheet property values.		✓	
3.5 Use header elements to convey document structure and use them according to specification.		✓	
3.6 Mark up lists and list items properly.		✓	
3.7 Mark up quotations. Do not use quotation markup for formatting effects such as indentation.			✓
6.5 Ensure that dynamic content is accessible or provide an alternative presentation or page.		✓	
7.2 Until user agents allow users to control blinking, avoid causing content to blink (i.e., change presentation at a regular rate, such as turning on and off).			✓
7.4 Until user agents provide the ability to stop the refresh, do not create periodically auto-refreshing pages.			✓
7.5 Until user agents provide the ability to stop auto-redirect, do not use markup to redirect pages automatically. Instead, configure the server to perform redirects.			✓
10.1 Until user agents allow users to turn off spawned windows, do not cause pop-ups or other windows to appear and do not change the current window without informing the user.		✓	

11.1 Use W3C technologies when they are available and appropriate for a task and use the latest versions when supported.		✓	
11.2 Avoid deprecated features of W3C technologies.	✓		
12.3 Divide large blocks of information into more manageable groups where natural and appropriate.	✓		
13.1 Clearly identify the target of each link.		✓	
13.2 Provide metadata to add semantic information to pages and sites.			✓
13.3 Provide information about the general layout of a site (e.g., a site map or table of contents).	✓		
13.4 Use navigation mechanisms in a consistent manner.	✓		
<b>And if you use tables (Priority 2)</b>	<b>Yes</b>	<b>No</b>	<b>N/A</b>
5.3 Do not use tables for layout unless the table makes sense when linearized. Otherwise, if the table does not make sense, provide an alternative equivalent (which may be a linearized version).		✓	
5.4 If a table is used for layout, do not use any structural markup for the purpose of visual formatting.	✓		
<b>And if you use frames (Priority 2)</b>	<b>Yes</b>	<b>No</b>	<b>N/A</b>
12.2 Describe the purpose of frames and how frames relate to each other if it is not obvious by frame titles alone.			✓
<b>And if you use forms (Priority 2)</b>	<b>Yes</b>	<b>No</b>	<b>N/A</b>
10.2 Until user agents support explicit associations between labels and form controls, for all form controls with implicitly associated labels, ensure that the label is properly positioned.		✓	
12.4 Associate labels explicitly with their controls.		✓	
<b>And if you use applets and scripts (Priority 2)</b>	<b>Yes</b>	<b>No</b>	<b>N/A</b>
6.4 For scripts and applets, ensure that event handlers are input device-independent.		✓	
7.3 Until user agents allow users to freeze moving content, avoid movement in pages.			✓
8.1 Make programmatic elements such as scripts and applets directly accessible or compatible with assistive technologies [Priority 1 if functionality is important and not presented elsewhere, otherwise Priority 2.]		✓	
9.2 Ensure that any element that has its own interface can be operated in a device-independent manner.		✓	
9.3 For scripts, specify logical event handlers rather than device-dependent event handlers.		✓	

Tabell 3: Testresultater fra manuell gjennomgang (prioritet 2)

**Sjekkpunkter, prioritet 3**

I Tabell 4 har jeg samlet punkter med prioritet 3. Dette er punkter som *kan* tilfredsstilles, nettstedet vil få en "AAA" rating dersom disse er tilfredsstilt. Tabell 4 har totalt 19 punkter, hvorav 14 er aktuelle for Altinn. Altinn tilfredsstiller her 4 av 14 aktuelle retningslinjer.

<b>In General (Priority 3)</b>	<b>Yes</b>	<b>No</b>	<b>N/A</b>
4.2 Specify the expansion of each abbreviation or acronym in a document where it first occurs.		✓	
4.3 Identify the primary natural language of a document.		✓	

9.4 Create a logical tab order through links, form controls, and objects.	✓		
9.5 Provide keyboard shortcuts to important links (including those in client-side image maps), form controls, and groups of form controls.	✓		
10.5 Until user agents (including assistive technologies) render adjacent links distinctly, include non-link, printable characters (surrounded by spaces) between adjacent links		✓	
11.3 Provide information so that users may receive documents according to their preferences (e.g., language, content type, etc.)		✓	
13.5 Provide navigation bars to highlight and give access to the navigation mechanism.	✓		
13.6 Group related links, identify the group (for user agents), and, until user agents do so, provide a way to bypass the group.		✓	
13.7 If search functions are provided, enable different types of searches for different skill levels and preferences.			✓
13.8 Place distinguishing information at the beginning of headings, paragraphs, lists, etc.		✓	
13.9 Provide information about document collections (i.e., documents comprising multiple pages.).		✓	
13.10 Provide a means to skip over multi-line ASCII art.			✓
14.2 Supplement text with graphic or auditory presentations where they will facilitate comprehension of the page.			✓
14.3 Create a style of presentation that is consistent across pages.	✓		
<b>And if you use images and image maps (Priority 3)</b>	<b>Yes</b>	<b>No</b>	<b>N/A</b>
1.5 Until user agents render text equivalents for client-side image map links, provide redundant text links for each active region of a client-side image map.			✓
<b>And if you use tables (Priority 3)</b>	<b>Yes</b>	<b>No</b>	<b>N/A</b>
5.5 Provide summaries for tables.		✓	
5.6 Provide abbreviations for header labels.		✓	
10.3 Until user agents (including assistive technologies) render side-by-side text correctly, provide a linear text alternative (on the current page or some other) for <i>all</i> tables that lay out text in parallel, word-wrapped columns.			✓
<b>And if you use forms (Priority 3)</b>	<b>Yes</b>	<b>No</b>	<b>N/A</b>
10.4 Until user agents handle empty controls correctly, include default, place-holding characters in edit boxes and text areas.		✓	

Tabell 4: Testresultater fra manuell gjennomgang (prioritet 3)



## 7.4. Diskusjon

Jeg vil i dette kapittelet fokusere på brukervennlighet for brukere med tilpassningsbehov; altså funksjonshemmede, svaksynte osv. Jeg har valgt å ikke ta opp andre brukervennlighetsaspekter, fordi det er en langt bredere diskusjon enn fokuset i dette kapittelet og dette tas opp andre steder i oppgaven (kapittel 5 og 6). Likevel er det verdt å merke seg at enkelte av temaene jeg diskuterer også vil kunne bedre bruksopplevelsen for brukere uten funksjonshemninger.

Jeg tar kun opp de punktene som *ikke* ble tilfredsstilt under testing. Punkter som *ble* tilfredsstilt, eller punkter som ikke er aktuelle for Altinn omtales ikke videre.

Diskusjonen er oppdelt i to deler. Jeg har samlet det jeg anser for de viktigste punktene i ett delkapittel kalt "primære forbedringspunkter". De resterende punktene kommer i "sekundære forbedringspunkter". Denne oppdelingen er basert på mitt eget skjønn, etter hva jeg mener burde endres ved Altinn for at løsningen skal fungere best mulig. De viktigste områdene er samlet i den første delen av diskusjonen, mens den siste halvdel tar opp punkter som jeg mener ikke er like kritiske å rette – selv om de selvfølgelig er viktige de også. I denne oppdelingen har jeg ikke lagt vekt på W3Cs prioriteringer, det vil si at du vil finne punkter med prioritet 2 og 3 under primære forbedringspunkter. Jeg kommenterer grunnen til min oppgradering av prioriteten under hvert enkelt punkt.

### Primære forbedringspunkter

Her tar jeg opp det jeg anser som viktigst å forbedre i Altinn. Dersom Altinn skal tilrettelegges for universell utforming burde man, etter min mening, begynne med disse punktene.

#### Støtte for grafiske nettlelere

Stort sett ser testresultatene i de grafiske nettleserne bra ut. Så lenge vi ikke tester for universell utforming fungerer Altinn godt i de testede nettleserne. Men når vi begynner å kjøre tester for universell utforming ser vi straks svakheter. Inspeksjon av kildekode viser at det blitt brukt visuelt orientert utvikling. Med det mener jeg at utviklingen av brukergrensesnittet ikke har tatt høyde for alternativ bruk av løsningen.

#### Støtte for ikke-grafiske nettlelere

Det var to krav som måtte tilfredsstilles for at støtte for ikke-grafiske nettlelere kunne bli tilfredsstilt.

1. Finnes det tilsvarende informasjon i en tekst- eller opplesende nettleser som i en grafisk nettleser?
2. Er informasjonen som presenteres presentert i en meningsfull rekkefølge når det leses serielt?

Punkt 1 tilfredsstilles for så vidt. Nesten all informasjonen er der, men som skjermdumpen av Altinn (Figur 40, side 126) viste er ikke informasjonen særlig leselig fordi Altinn feiler på punkt 2.

Dette gjør Altinn svært vanskelig, hvis ikke umulig, å bruke for blinde og andre som bruker rent tekst-baserte nettlesere. De enkelte nettsidene er ikke kodet logisk. Resultatet er at det er vanskeligheter med navigering, og det er nesten umulig å gjøre det mest sentrale i Altinn; fylle ut skjemaer.

I Figur 41 har jeg tatt med en skjermdump av et nettsted med velformet HTML-kode. Her ser vi at overskriftene er tydelig markert. Alle tekstbaserte nettlesere har mulighet til å hoppe mellom forskjellige overskrifter, og også mellom linker. Det betyr at brukere kan slippe trykke seg gjennom unødvendig informasjon. Enda viktigere er dette for blinde/svaksynte som bruker høytlesende nettlesere. Dersom de ikke kan hoppe rundt på siden uten å høre gjennom all teksten på nettsiden blir siden mer eller mindre ubrukelig for dem.



**Figur 41: Skjermdump av nettsted med velformet innhold (Opera – tekstmodus)**

Nå har det allerede blitt tatt en avgjørelse om at tekst-baserte nettlesere ikke skal støttes (jf. Hvilke nettlesere som støttes av Altinn, se side 13). Men like fullt er det mange som bruker høytlesende nettlesere som bruker Explorér som motor. Disse vil ha nytte av en tjeneste som er bygget opp på en fornuftig måte.

### **Alternativ tekst**

Dette er et punkt med prioritet 1 etter WCAG. Altinn mangler nesten gjennomgående såkalt alternativ tekst. Forsidens "logg inn"-knapper er et unntak. At et bilde har alternativ tekst betyr at utviklerne legger inn en tekstlig beskrivelse av bilder og grafikk som brukes i nettløsningen. For eksempel er begge logg inn-

knappene i Altinn laget som grafikk. Uten alternativ tekst på disse knappene hadde det vært umulig å vite hvilken knapp man skal trykke på når grafikk er slått av i nettleseren (se Figur 42.)



**Figur 42: Innlogging i Altinn uten grafikk (Opera)**

W3C har svært strenge krav for alternativ tekst; *all* grafikk skal ha tekstlig beskrivelse i følge retningslinjene: "[...] This includes: images, graphical representations of text (including symbols), image map regions, animations (e.g., animated GIFs), applets and programmatic objects, ascii art, frames, scripts, images used as list bullets, spacers, graphical buttons, [...]" W3C / WAI 1999.

At t.o.m. liste bullets (selve punktene i en punktliste) og "spacers" (usynlig grafikk som benyttes til å "dytte" elementer i en nettside fra hverandre) skal beskrives er strengt. Det strides en del om hvorvidt dette er nødvendig, men WCAG er helt klart på dette. I Altinns tilfelle betyr det f.eks at all grafikken som brukes i skjema-navigeringen for å vise hvor brukeren er også må ha tekstlig beskrivelse. Enkelte eksperter og svaksynte brukere mener at det er bedre å legge inn tomme tekstlige beskrivelser på slik grafikk. Deres argumenter er at det må være bedre for blinde å svaksynte å slippe å høre tekstleseren deres lese opp fraser som "spacer" og "bullet", siden dette er elementer som brukes for *visuell layout*. Argumentet går derfor slik at det kun er grafikk som har *innholdsmessig* verdi som skal ha tekstlige alternativer.

Jeg er enig med de som mener at det er dumt å beskrive *alle* grafiske elementer. Men jeg mener at det er en bedre måte å løse dette på. Ved å kode korrekt (velformet) HTML, og bruke CSS til layout og presentasjon kan man rense HTML-koden for mye informasjon som skal brukes til presentasjon. Idealet er at HTML-dokumentene inneholder selve innholdet, mens presentasjon skilles ut i andre dokumenter, som CSS. Her har Altinns implementasjon stort forbedringspotensiale.

Det er ikke bare svaksynte og blinde som kan ha nytte av å ha grafikken avslått. Dersom man slår av grafikk kan internettsurfing være markant raskere, dette er nyttig for folk som har trege linjer. I tillegg blir det stadig flere som surfer på mobiltelefoner, eller bruker mobiltelefonen som oppringt forbindelse til en bærbar datamaskin. Forbindelse over mobil er fortsatt tregt, men enda viktigere: dyrt. Siden du betaler for det du laster ned, vil mange være interesserte i å kutte ut "unødvendig" nedlasting.

**Korrekt koding av forms**

Dette er et prioritert 2 punkt i WCAG. Retningslinjene 10.2 og 12.4 sier at tekstfelter og deres tilhørende navn skal stå samlet..

”Forms” er begrepet som brukes om byggestene for å bygge skjema i HTML. Enhver nettside eller applikasjon som bruker forms må kodes slik at tekstfelter og deres tilhørende navn står samlet. Dette er ikke tilfelle i Altinn, som vi kan se av skjermdumpen på side 126.

Dette punktet hører under det å følge HTML-standarder når man koder nettsider. Jeg tar opp dette punktet her, fordi dette er den aller viktigste grunnen til at sidene er ubrukelige for blinde. Dersom en blind bruker mot formodning skulle ha nok tålmodighet til å komme seg inn på en skjemaside vil det være *umulig* for han å fylle ut skjemaet. Dette burde være det første som fikses i Altinn dersom løsningen skal tilfredsstille krav til universell utforming.

**Bruk uten CSS**

Dette er prioritert 1 i WCAG. Punkt 6.1 sier at dokumenter må kunne leses uten bruk av stilark (CSS).

CSS er for det meste brukt på en hensiktsmessig måte, men den underliggende dokumentstrukturen er ikke bra. Dokumentene er ikke ordnet logisk og elementene i skjemaene har ikke blitt koplet sammen. Resultatet er at dersom man slår av CSS, vil sidene være umulige å bruke. Et eksempel kan ses i Figur 40 på side 126.

**Justering av tekststørrelse**

Å kunne justere tekststørrelsen på nettsider er viktig for mange. At brukere selv kan endre tekststørrelsen i nettleseren er kanskje ikke noe den gjennomsnittlige bruker er klar over. Svaksynte kjenner nok til dette, men for middelaldrende mennesker og eldre er det mange som ikke er klar over at nettlesere tilbyr mulighet for å øke skriftstørrelsen.

De fleste nye nettlesere (f.eks. Opera, Firefox, Netscape) har støtte for dette på en måte som gjør at det er uvesentlig hvordan tekststørrelsen er implementert i nettsidene. Nettleseren tar ansvaret for å øke tekststørrelsen uavhengig av implementasjonen.

Operas nettleser tar dette et trinn lenger og skalerer opp *alt* dersom man bruker ”zoom”-funksjonen. Både grafikk og mellomrom blåses opp proporsjonalt, slik at hele layouten forblir den samme, bare i større proporsjoner.

Dessverre er det slik at den nettleseren som har størst fotfeste ikke har denne muligheten. Microsoft Internet Explorer klarer ikke å skalere opp tekststørrelsen på et nettsted dersom tekststørrelsen er kodet med såkalte absolutte verdier (f.eks. piksler, punkter eller millimeter). Man må kode med relative størrelser (”em”,

prosent, nøkkelord ("smaller" / "bigger" osv.)) for at Explorer skal klare å skalere opp teksten.

Det er ikke bare Explorer som trenger relative størrelser. Mobiltelefoner og PDAs er også enheter med små skjermer; disse trenger også relative tekststørrelser å forholde seg til. Å vise 12 punkters tekst på en vanlig PC-skjerm er ikke noe problem. Å vise 12 punkters tekst på en liten PDA skjerm betyr mye bortkastet plass. I alle tilfeller burde det være opp til brukeren hvor stor teksten skal være.

Nå skal det sies at Altinn faktisk har et oppsett for svaksynte. Under "min profil" kan brukeren krysse av for "Bruk skjema for synshemmede". Løsningen vil da bytte til et oppsett der tekststørrelsen er blåst opp. Dette gjelder likevel kun i selve skjemateksten. Det gjelder ikke resten av nettstedet, og heller ikke i navigeringsmenyen i skjemaet. Det virker litt halvgjort, og dette kunne også vært implementert på en langt bedre måte. Hvordan skal synshemmede finne dette valget? Jeg tror ikke så mange går inn innstillingene. Spesielt kanskje ikke "vanlige" brukere som kommer for å løse en oppgave; ikke for å utforske programvaren.

### **Bruk uten mus**

Både Explorer og Opera håndterte Altinn bra med tastatur. I Explorer fungerer det meste av utfyllingen og selve navigasjonen innenfor en skjemaside godt.

Opera har som nevnt bedre kontroller for å surfe med tastatur; her finnes mange av de samme mulighetene som finnes i tekstbaserte nettlesere: mulighet for å hoppe mellom overskriftene på en side, bare hoppe mellom linker, hoppe mellom lister osv.

I tillegg til bedre muligheter for navigering har Opera også bedre tilbakemelding, nettleseren markerer alle linker man hopper mellom slik at man ser hvilken link som er valgt. Dette er noe *nettleseren* gjør. Utviklerne av Altinn har gjort slik at noen av linkene i Altinn utheves når man navigerer til en link med tastaturet. Dette er dessverre ikke gjort i skjema-navigasjonen. Dette gjør det umulig å vite hvilken link som er aktiv i Explorer under tastatur-navigasjon. Dessverre er heller ikke statuslinjen i nettleseren behjelpelig fordi linken som vises ikke gir noen mening. Et eksempel: "javascript: \_\_doPostBack('FF\$nM\$\_ctl127'm')".

Dersom man surfer med mus vil man være så heldig å få en tekstboble som forteller deg hvor linken peker (i tillegg til tilbakemeldingen ved å holde musepekeren over selve linken). De som surfer med tastaturet i Explorer er dessverre ikke like heldige. Dette gjør at prøv-og-feil metoden er den eneste muligheten. Eventuelt må man surfe med "forrige" og "neste"-knappene, noe som er fullt mulig, men svært frustrerende dersom man ikke fyller ut skjemaet strengt serielt.

Jeg nevnte at Opera har muligheter for avansert side-navigering med tastaturet, man kan hoppe mellom overskrifter, lister og linker. Dette er svært praktisk og tidsbesparende dersom man bruker tastaturet til å surfe. Dessverre fungerer ikke denne muligheten i Altinn. HTML-sidene er ikke kodet slik standardene er lagt opp. Overskrifter er ikke kodet som overskrifter, lister er ikke kodet som lister osv. Dette

har innvirkning på bruksopplevelsen for de som bruker tastaturet. Dette er kanskje ikke så viktig dersom man bruker en grafisk nettleser, men det er likevel mange mennesker som foretrekker å bruke tastatur fremfor mus, ikke minst på grunn av musesyke og slitasjeskader. Da er det en fordel å kunne slippe å klikke seg forbi alle lenkene på en nettside ved å hoppe mellom seksjoner.

### **Bruk av scripts og dynamiske løsninger**

*8.1 Make programmatic elements such as scripts and applets directly accessible or compatible with assistive technologies [Priority 1 if functionality is important and not presented elsewhere, otherwise Priority 2.]*

Det er klart at funksjonalitet er svært viktig i Altinn. Derfor burde denne funksjonaliteten gjøres tilgjengelig for flest mulig. Uten funksjonaliteten er det jo ingen applikasjon å bruke. Her foreslår W3C at man gjør script-invokasjonen uavhengig av mus, slik at funksjonaliteten også fungerer dersom man bruker tastatur. Ellers er det også viktig å vite hvilken funksjonalitet som skal kjøres på klienten og hvilken som skal kjøres på tjeneren. Her er det som tidligere nevnt mange avveininger, men det er viktig å ta hensyn til brukere med andre behov når man gjør disse avveiningene.

*6.3 – Ensure that pages are usable when scripts, applets, or other programmatic objects are turned off or not supported. If this is not possible, provide equivalent information on an alternative accessible page.*

Dette punktet sier i bunn og grunn at en nettside skal være brukbar uten bruk av script i klienten/nettleseren. Dette er et punkt som jeg tror er vanskelig å tilfredsstille for en webapplikasjon. Spesielt i en så komplisert løsning som Altinn. Det *er* mulig å gjøre alle beregninger på tjeneren, men dette er sjelden hensiktsmessig. Her kommer mange avveininger inn, men å gjøre alle slike beregninger på tjeneren betyr som regel at bruksopplevelsen degraderes fordi brukeren ender opp med å vente på at tjeneren skal svare. Dessuten betyr det større belastning på tjeneren og linjekapasiteten siden det må sendes mer informasjon fram og tilbake mellom klient og tjener. I tillegg ødelegges bruksopplevelsen fordi tilbakemeldinger ikke kan være umiddelbare, noe som ødelegger konsentrasjonen til brukere.

### **Korrekt bruk av kodestandarder**

Dette er punkter med prioritet 2 i WCAG.

3.1 Bruk korrekt markup istedet for grafikk til å formidle informasjon.

3.2 Lag dokumenter som validerer til standardene.

3.3 Bruk stilark for layout og presentasjon.

3.4 Relative størrelser.

3.5 Bruk header-elementer og struktur.

3.6 Bruk liste-elementer.

11.1 Bruk W3C teknologier når mulig, og bruk de riktig og hensiktsmessig.

11.2 Ikke bruk utdaterte teknikker.

Jeg har nevnt noe av dette allerede under "Korrekt koding av forms" og "Bruk uten CSS". Grunnen til at jeg anser dette som et kritisk punkt er at løsningen vil fungere langt bedre for de med behov for tilpassning dersom kodestandardene følges. I tillegg vil også Altinn fungere i flere nettlesere dersom man følger kodestandardene. Et resultat av å følge standardene er at mange av problemene som er kommet fram i denne rapporten løses. Derfor vil det være hensiktsmessig å ha dette punktet som utgangspunkt.

### **Sekundære forbedringspunkter**

#### **Endring av vindusstørrelse og skjermoppløsning**

Altinn fungerer optimalt i 1024x768, dette tilsvarer oppløsningen til 17 tommeres bilderørskjermer eller 15 tommeres LCD-skjermer. Eventuell høyere oppløsning brukes ikke i det hele tatt. Dersom du kjører lavere skjermoppløsning får brukeren horisontal skrollbar ganske fort. Løsningen er nok brukbar i 800x600, men brukeren må da skrolle for å se hjelpetexten. Dette kan gå, avhengig av hvor ofte brukeren trenger å se i hjelpetexten.

Oppsettet i brukergrensesnittet er basert på ELMER-prinsippene. Løsningen har derfor tre kolonner, noe som tilsvarer mye horisontal skjerm plass. De fleste i dag har skjermer med høy oppløsning (1024 eller høyere). Det skal derfor ikke være noe problem å kreve så mye skjerm plass. Problem oppstår derimot fordi en del mennesker med synsproblemer kjører en lavere skjermoppløsning enn det skjermen er beregnet for. De gjør dette enten fordi de har problemer med å lese teksten, eller fordi de synes det er mer behagelig med stor tekst på skjermen. Derfor kjører de kanskje 800x600 på en skjerm som er beregnet for 1024x768. Dette gjelder ikke bare eldre mennesker, selv folk i 20-årene gjør dette.

Det at grensesnittet ikke skalerer etter oppløsningen skyldes nok problemene som oppstår ved å lage såkalt flytende layouts. De er svært kompliserte å få til å virke pent på alle plattformer og nettlesere. Det er enklere å sette en nedre grense for oppløsning og holde seg til den.

Den mest umiddelbare ulempen er at skjemaene inneholder så mye informasjon at det er lett å miste oversikten. En av fordelene med papirskjema er at du har hele skjemaet rett foran deg og du har overblikk. I en elektronisk løsning virker det mer "klostrofobisk" fordi du "stanger" i veggene hele tiden.

### **Språk**

Punkt 14.1 i WCAG sier at språket skal være enklest mulig. Jeg har overfladisk vurdert dette. Personlig synes jeg ikke at det er komplisert språk, men ofte svarer ikke hjelpeteksten på det jeg lurte på. Man leser ca 25 % saktere på skjerm enn på papir [Nielsen 1999a: 101], man burde derfor prøve å forenkle teksten enda mer. Eventuelt legge til rette for papirutskrifter av hjelpen, slik at man kan slå opp. Dette er en komplisert problemstilling, og heller ikke spesielt lett å måle.

Jeg ser for meg en løsning der den umiddelbare hjelpeteksten man får opp er kort og konsis. Dersom man fortsatt trenger hjelp kan brukeren rutes til en mer fyldig hjelpetekst (f.eks. papirutgaven) og lese den. Dette dokumentet burde være lett å skrive ut, slik at brukeren kan ha hjelpeteksten ved siden av seg mens han bruker Altinn.

### **Tabellbasert layout**

Tabell-basert layout var den tradisjonelle måten å lage nettsider på. Men med utbredt støtte for stilark er ikke dette like nødvendig lenger. Det brukes lite tabeller i Altinn, faktisk kun på et overordnet nivå for å dele inn i de tre kolonnene som finnes. Dette er ikke så ille, og trenger strengt talt ikke fikses. På den annen side; dersom sidene skal følge WCAG-standardene skal ikke tabeller brukes til layout.

### **Tilgjengelig versjon**

Punkt 11.4 sier følgende: Hvis det ikke er mulig å lage en tilgjengelig versjon, tilby en alternativ side som *er* tilgjengelig.

Dette er ansett som en siste utvei, dersom man ikke klarer å lage den "vanlige" versjonen tilgjengelig. Dette er et omstridt punkt. Mange mener det å ha to versjoner å oppdatere fører til at den ene (tilgjengelige versjonen) dermed vil være utdatert. Den versjonen som får mest oppmerksomhet vil alltid være den normale. I tillegg får man plutselig to versjoner å forholde seg.

Altinn tilbyr som tidligere nevnt et systemvalg for svaksynte som skal hjelpe de. Men det eneste den tilbyr er større skrift enkelte steder. Den er ikke mer tilgjengelig enn den normale versjonen dersom man er blind. Jeg mener det er mulig å lage én variant som er langt bedre enn dagens versjon for svaksynte. Ved å bygge om normalvarianten av Altinn kan den tilby en bedre bruksopplevelse for både svaksynte og mennesker med normalt syn.

### **Bruk av nye vinduer**

Punkt 10.1 går på bruk av nye vinduer. Det åpnes en del nye vinduer i løsningen. Spesielt gjelder dette for hjelpe-sidene. Å åpne sider i nye vinduer er noe utviklere bør tenke nøye gjennom. Den gjennomsnittlige bruker merker som regel ikke at en link blir åpnet i nytt vindu, spesielt ikke dersom han surfer i maksimert vindu [Nielsen 1999b]:

*"Users often don't notice that a new window has opened, especially if they are using a small monitor where the windows are maximized to fill up the screen. So a user who tries to return to the origin will be confused by a grayed out Back button."*

Dermed vil et nytt vindu bare legge seg på toppen. Dette ødelegger for bruk av "back"-knappen; den nest mest brukte av navigasjonsmekanismene på nettet [Nielsen 1999b].



Den gjennomsnittlige "poweruser" blir som regel irritert når et nettsted åpner et nytt vindu. Eller som Nielsen sier det [Nielsen 1999b]:

*"Opening up new browser windows is like a vacuum cleaner sales person who starts a visit by emptying an ash tray on the customer's carpet. Don't pollute my screen with any more windows, thanks (particularly since current operating systems have miserable window management). If I want a new window, I will open it myself!"*

WCAG legger vekt på at bruk av nye vinduer skal unngås. Det er likevel tilfeller der det kan være hensiktsmessig å åpne nye vinduer. Dersom utviklerne mener at bruk av nytt vindu er nødvendig, burde dette gjøres klart for brukeren før han trykker på linken.

Et annet moment er at det etterhvert er blitt populært med såkalte "popup-blockers". Disse brukes primært til å hindre at man blir nedrent av reklame på nettsider som åpner vinduer uten at man ber om det. Noen av disse popup-blokkene er enkelt implementert og blokker *alle* nye vinduer som åpnes. De fleste nye slike blokkere tar hensyn til linker som brukeren selv klikker på. Men det finnes fortsatt folk som bruker gamle popup-blokkere.

Alle nettlesere har mulighet for å åpne linker i nytt vindu. Ideelt sett burde dette valget legges i *brukerens* hender, ikke i utviklerens. Problemet er at den gjennomsnittlige bruker ikke vet om denne muligheten (den finnes som regel ved å trykke høyre musknapp (Windows, Linux), eller ved å "option-klikke" (Mac) på en link.

Jeg mener dette valget burde legges i brukerens hender. Så lenge back-knappen fungerer er det lettere for brukeren å forholde seg til ett vindu. Her er det en del tekniske problemer som kan oppstå, spesielt med tanke på webskjema og lagring. Men det er viktig å være klar over at mange brukere synes det er forvirrende med flere vinduer. De trykker på en link (som åpnes i nytt vindu uten at de legger merke til det), leser teksten der, og prøver å trykke på "back"-knappen – men den fungerer ikke! Og der stopper de rett og slett opp. Det er ikke før de velger å lukke vinduet at de finner nettstedet der de var. Slike ting skaper frustrasjoner.

### **Oppdeling av informasjon i mindre bolker**

Å dele opp informasjon på en nettside i mindre bolker som er mer fordøyelig er viktig for alle. Det gjør det lettere å scanne sidene, og det gjør det lettere å finne tilbake til det du holdt på dersom du mister tråden eller blir forstyrret. Altinn er godt visuelt oppdelt, med klare seksjoner og felter. Denne oppdelingen finnes ikke i kildekoden, og dermed heller ikke for de som bruker tekstbaserte nettlesere eller nettlesere med talesyntese. Som jeg har nevnt tidligere kan dette rettes ved å bygge om sidene slik at de kodes etter W3Cs standarder.

**Klargjør hvor linker peker**

Punkt 13.1 i WCAG påpeker at det er viktig for en bruker å vite hvor en link peker uten å lese konteksten. Som tidligere nevnt scanner de fleste mennesker nettsider, de leser ikke all teksten nøye. De hopper som regel fra link til link. Det betyr at linker med teksten "klikk her" ikke gir noen mening. Dersom en linktekst består av "klikk her" må de derfor hoppe litt tilbake og lese teksten før. Da hadde det vært bedre å bruke en linktekst som beskriver hva linken peker til. Det samme gjelder i enda større grad for blinde som gjerne hopper mellom linkene på en side i en talende nettleser.

I Altinn kan dette illustreres i hovedsiden. Dersom man har flyttet flere skjema til hovedsiden ligger de i en oversikt, med flere valg for hvert skjema: "Åpne", "Vis", "Om skjema" og "Slett". For brukere med normalt syn vil det ikke være mye jobb å følge linjene i tabellen og se hvilket skjema det er snakk om. Scanning-oppførselen gjelder ikke i like stor grad i setninger som denne. Men for blinde vil det innebære å hoppe tilbake i teksten på nettsiden for å finne ut hvilket skjema det er snakk om. Men dette tar lenger tid fordi datamaskinen må lese opp teksten. De har ikke noen mulighet til å trekke en linje med øyet.

Det er mulig at det går an å legge inn "title"-elementer på linkene, slik at man beholder den strippede teksten til vanlig, men slik at en talende nettleser leser opp mer. For eksempel kan en taleleser i stedet for å lese opp "Yoga Ram" (under kolonnen "Sist endret av"), lese opp "Sist endret av Yoga Ram". Hvorvidt dette lar seg gjennomføre har jeg ikke fått testet.

**Andre momenter (prioritet 3)****Forkortelser**

Punkt 4.2 sier at første gangen en forkortelse brukes, bør forkortelsen kodes med <abbr> . Dette gjøres ikke idag. Dette er mest viktig for blinde, men det er også nyttig for de som ikke vet hva en forkortelse står for. Dersom man definerer forkortelsen i koden vil dagens nettlesere lage en liten tekstboble når man holder peker over forkortelsen. I tekstboblens vil forkortelsen stå i fulltekst. For blinde betyr det at forkortelsen vil leses som en forkortelse og ikke som et ord.

**Identifikasjon av språk**

Punkt 4.3 Identifiser språket på sidene.

Akkurat som en forkortelse må markeres for at en datamaskin forstår at det er en forkortelse, må språket til en nettside identifiseres. Dette er nok en gang viktigst for blinde, hvertfall i nettapplikasjoner. Det kan argumenteres at en blind person kan sette nettleseren sin til det språket siden er på, men ved å definere språket i kildekoden vil nettleseren automatisk kunne bytte til korrekt talesyntese når den kommer inn på siden.

### **Snarveier via tastaturet**

9.5 Tilby snarveier via tastaturet.

Det er lagt inn tastatur-snarveier i Altinn. Dette kan være kjekt å bruke for mennesker uten funksjonshemninger, og svært nyttig for de som bruker tastaturet. Dessverre var det ikke lett å finne disse snarveiene. Jeg oppdaget snarveiene da jeg gjorde en inspeksjon av kildekoden. Det ser ikke ut til at de er spesielt godt markert i applikasjonen. Jeg antar at de står i hjelpe-teksten, men de burde også markeres i selve applikasjonen.

### **Tilby dokumenter/nettssider basert på brukers ønsker**

Punkt 11.3 sier følgende: Provide information so that users may receive documents according to their preferences (e.g., language, content type, etc)

Her er det snakk om å legge inn muligheter for at folk kan komme til den versjonen av et dokument som de vil ha. Et aktuelt tilfelle er å legge inn bokmål og nynorsk på en slik måte at en nynorsk bruker vil få servert Altinn i nynorsk drakt automatisk. Dette forutsetter at brukeren har satt dette valget i nettleseren sin.

Et annet valg under dette punktet er å ha flere typer stilark (CSS) alt etter hva slags enhet man bruker. F.eks. kan brukere av PDAer få servert en annen versjon av Altinn som er optimalisert for skjermoppsettet på en PDA eller mobiltelefon.

### **Gruppering av relaterte linker**

Navigasjonselementer burde markeres som en gruppe i kildekoden. I tillegg burde det tilbys en link som kan hoppe over navigasjonen. Denne linken er kjærkommen for blinde, som dermed kan slippe å høre på en lang liste med linker som kanskje er uninteressante; spesielt dersom de er ute etter innholdet på siden.

### **Plassér utskillende informasjon i begynnelsen av strukturelle elementer som overskrifter og lister**

Dette punktet tilfredsstilles ikke fordi dokumentene ikke er kodet korrekt. Men det henspiller også på å så tidlig som mulig uttale poenget med en setning eller et avsnitt. Dermed vil både brukere som "scanner" siden og blinde som "skimmer" (den blinde varianten av "scanning") lettere vite om det er dette de leter etter, eller om de skal lete videre på siden.

## 7.5. Oppsummering

Dette kapittelet har tatt for seg hvordan Altinn håndterer brukere som har behov for tilrettelagte tjenester, for eksempel svaksynte og blinde. Systemet er blitt testet i automatiske verktøy og deler av kildekoden er blitt gjennomgått. Resultatene viser at Altinn ikke tilfredsstiller retningslinjer for universell utforming. En analyse er blitt presentert der de viktigste funnene er gjennomgått og hva som må gjøres for at Altinn skal tilfredsstille krav til universell utforming.

For å komme opp på nivå A må følgende gjøres:

- Sidene må kodes om slik at dokumentstrukturen er logisk
- Sidene må kunne leses uten CSS
- Det må legges inn alternativ tekst på grafikk og bilder
- Bruk av javascript må eksekveres riktig og også håndtere bruk uten mus

For å komme opp på nivå AA må først punktene for nivå A tilfredsstilles. Deretter må 16 ytterligere punkter utbedres. Disse punktene finnes på side 130, jeg lister opp de viktigste her:

- Forms må kodes korrekt
- Det bør brukes relative størrelser på fonter, marger og padding
- Ikke åpne nye vinduer uten å gjøre det klart for brukeren på forhånd
- Klargjør hvor linker peker

For at Altinn skal komme opp på nivå AAA må punktene under A og AA først tilfredsstilles. Deretter må ytterligere 10 punkter fikses, jeg angir noen her, resten finnes på side 131:

- Legg inn fyll-tekst i tomme form-elementer
- Relaterte linker må grupperes
- Spesifiser forkortelser
- Angi språket i dokumentet
- Tilby sammendrag av tabeller
- Gruppér relaterte dokumenter / dokumentsamlinger

## 8. KONKLUSJON

### 8.1. Introduksjon

Altinn startet som en alternativ innrapporteringskanal for næringslivet. Den nye trenden er at Altinn vil være sentral i å tilby elektroniske tjenester til både næringsliv og borgere. Dette stiller nye krav til en løsning som i utgangspunktet skulle håndtere innrapportering av data til myndighetene fra næringslivet. I dag skal Altinn bidra til bedre kommunikasjon mellom det offentlige og det private. Løsningen dekker ikke bare næringslivet, den dekker også privatpersoner. Dette gjør inntoget av sporadiske brukere enda større enn det var da Altinn kun dekket næringslivet.

#### Forbedringer fra to sider

Som vi så i kapittel 4.4 (side 45) er ansvaret for brukeropplevelsen i Altinn todelt mellom Altinn sentralforvaltning og etatene. Både portalen som sådan og selve skjemaene bør forbedres. Dette betyr at både etatene og ASF må ta tak i hver sine deler.

Dagens skille mellom portal og skjema er for skarp. Brukeren har problemer med å se forskjellen på Altinn og på den etaten han fyller ut skjema til. Som en konsekvens av dette bør man se om det er mulig å viske ut litt av skillet mellom portal og skjema. Dette kan spesielt hjelpesystemet nyte godt av.

### 8.2. Forbedringer i webportalen

Det er mange områder i webportalen og rammeverket som kan dra nytte av forbedringer i brukeropplevelsen. Forbedringer i disse områdene vil være med på å heve brukeropplevelsen i løsningen som helhet.

#### Findability

For å øke finnbarheten til skjema er det flere ting som kan gjøres.

Skjemaorganiseringen er i dag "gjemt" bak sikkerhetsløsningen, og organiseringen av katalogen er avsenderorientert, ikke brukerorientert. Det er også vanskelig å søke på skjema.

Forslag for å løse dette:

- Skjemakatalogen bør tilby alternative sorteringsmåter, for eksempel etter brukergrupper, emner og oppgaver.
- Det bør gå an å søke på metadata til skjema, ikke bare i tittel.
- Skjemakatalogen bør ligge utenfor sikkerhetsløsningen slik at det går an å lenke til spesifikke skjema fra andre nettsteder.

## Navigasjon og oversikt

Venstremenyen gir ikke nok oversikt og navigasjonsstøtte. For å løse dette kan følgende gjøres:

- La venstremenyen flyte sammen med skjemaet.
- Åpne for at lenker kan gå over flere linjer
- Fjerne den automatiske kuttingen av teksten i menyen.
- Mulighet for å gruppere lenker og bruke overskrifter på disse gruppene av lenker.

## Håndtering av vedlegg

Håndteringen av vedlegg og vedleggstabeller er vanskelig i dagens Altinn. Vedlegg er vanskelig å finne og å administrere. Vedleggene administreres via et annet skjema, noe som kompliserer skjermbildet i skjemaet. For å løse dette kan følgende gjøres:

- Innføring av "mapper", slik at dagens hovedskjema vises på lik linje med dagens vedleggsskjema. Det vil da ikke lenger være noe som heter vedleggsskjema.
- *Alle* skjema finnes da i skjemakatalogen og vil være søkbare.
- Brukeren kan organisere alle skjema i en leveranse fra den aktuelle "mappa". Mappa blir grensesnittet for administrering av skjema innen en leveranse, for eksempel selvangivelsen.
- Fjerne vedleggstabeller og i stedet bruke dynamiske tabeller.

## Lagring av brukers arbeid

Lagring av brukers arbeid omfatter flere punkter. For å bedre ta vare på arbeidet til brukeren kan følgende gjøres:

- Dialogbokser ved lagring bør få lik logikk, slik at brukeren kan ta nytte av gjenkjenningseffekten og klikke på samme valg.
- I dialogbokser for lagring bør "ok" være det valget som lagrer. Ved "avbryt" bør brukeren kunne gå ut av skjemaet uten å lagre.
- Ved timeout bør systemet lagre skjemaet, deretter bør systemet vise innloggingssiden. Denne siden bør vise en tekst som forklarer hva som har skjedd. Når brukeren så logger inn igjen bør systemet hente fram den siden brukeren var på før utlogging ved timeout.
- Det er uklart hvor ofte lagre-knappen nederst i skjermbildet skal brukes. Lagre-knappen kan døpes om til "Pause". Samtidig kan det innføres automatisk lagring i bakgrunnen via Ajax-teknikker. Dermed vil systemet håndtere byrden med å lagre, samtidig som brukerens behov for å legge fra seg skjemaet er opprettholdt. Oppførselen til "pause"-knappen bør være lik som ved timeout.

## Hjelp

De fleste problemene i hjelpesystemet skyldes Altinn delte organisering. For brukerne er det vanskelig å skille mellom Altinn sentralt og etaten som har skjemaet.

- Det er fire liknende valg når brukeren leter etter hjelp. Fire valg beholdes, men med én endring i navnene på hjelpevalgene: "Hjelp", "Hjelpetekst", "Kontakt oss" og "Om skjema".

- Det er vanskelig å finne rettledningen til skjema. I tilfeller der det refereres til rettledningen bør det lenkes direkte til den. Rettledningen kan fortsatt ligge på "Om skjema"-siden, men bør ligge øverst og være visuelt uthevet fra brødteksten.
- Det er vanskelig å finne kontaktinformasjon til faglig hjelp for utfylling av skjema. "Kontakt Altinn"-lenken bør endres til "Kontakt oss". På denne siden bør det ligge kontaktinformasjon til både Altinn teknisk brukerstøtte, og til faglig brukerstøtte. Ved å løfte opp faglig brukerstøtte nærmere brukeren bør det bli enklere for brukerne å finne denne informasjonen.
- Veiviseren i informasjonsportalen kommer ikke opp i en del moderne nettlesere. Hvorfor dette skjer bør bli avklart. Deretter bør det bli utbedret.

### Feilretting

Feilretting er en viktig del av innsendingsprosessen for skjema. Også her er det en del forbedringspunkter:

- Når man retter en feil og har feilflippen åpen går det ikke an å klikke på hjelpeikoner. Dette kan rettes ved at det blir mulig å klikke på hjelpeikonene direkte uten at det kommer opp dialogbokser som det gjør i dag.
- Detaljer om feil kommer opp som mouseover-tekst. Denne kuttes i enkelte nettlesere og er vanskelig å lese når teksten er lang fordi teksten forsvinner etter noen sekunder. Løsningen er å la hele feilmeldingen stå som vanlig tekst i feilfeltet.
- Låste felter er vanskelig å skille fra vanlige felter i feilsøkesituasjoner. Ved å endre det visuelle utseende på låste felter kan det bli lettere å se forskjell på vanlige felter og låste felter.

### 8.3. Forbedringer i skjemaautforming

Etatene har ansvaret for skjemaautformingen av sine egne skjema. Først og fremst har de ansvaret for informasjonsarkitekturen i skjemaene, men de har også ansvar for hjelpetekster og funksjoner som kalkuleringer.

Den heuristiske evalueringen fant noen forbedringspunkter i skjemaautformingen:

- Ved å legge inn avklaringsspørsmål (sporvalg) vil brukere få færre sider med urelevante spørsmål.
- Det er fortsatt en del steder i skjema der det ikke er lagt inn kalkuleringer. Det bør legges inn kalkuleringer alle steder der dette er mulig.
- Når man fyller ut skjemaet i tilfeldig rekkefølge er det av og til vanskelig å skjønne at enkelte felter henter data fra tidligere felter i skjemaet. I disse tilfellene må det gjøres klart at data hentes fra andre steder. Dette kan gjøres ved å legge inn tekst ved det aktuelle feltet som forteller hvor det hentes data fra.
- Mye av hjelpeteksten gir for lite hjelp i forhold til det brukeren lurte på i utfyllingsøyeblikket. Hjelpetekstene må skrives om for web. Konkret betyr dette å gjøre tekstene kortere og mer spisset i forhold til det brukerne lurte på i utfyllingsøyeblikket. Tekstene må også splittes opp og benytte overskrifter, lister og formattering. Dette er ikke mulig i dagens tekniske løsning.

Skjemastrukturen kan gjøres bedre og mer oversiktlig ved hjelp av sporvalg og restrukturering av innhold. Ved å fjerne postnumre og korte ned overskriftene i menyen kan menyen gjøres mer oversiktlig.

## 8.4. Universell utforming

Altinn er både kontraktsmessig og politisk pålagt å følge WCAG retningslinjene til WAI, men det står ikke noe om på hvilket nivå løsningen skal legge seg på.

Ut i fra resultatene av testene av Altinn ser vi at løsningen feiler på selv basale krav for universell utforming. Nivå A tilfredsstilltes ikke, og dermed heller ikke AA, eller AAA.

For at Altinn skal kunne sies å tilfredsstillte krav til universell utforming *må* løsningen i det minste imøtekomme kravene markert med prioritet 1 (nivå A). I dag er det 2 av 6 punkter som godkjennes.

For å komme opp på nivå A må følgende gjøres:

- Sidene må kodes om slik at dokumentstrukturen er logisk
- Sidene må kunne leses uten CSS
- Det må legges inn alternativ tekst på grafikk og bilder
- Bruk av javascript må eksekveres riktig og også håndtere bruk uten mus

For å komme opp på nivå AA må først punktene for nivå A tilfredsstilltes. Deretter må 16 ytterligere punkter utbedres. Disse punktene finnes på side 130, jeg lister opp de viktigste her:

- Forms må kodes korrekt
- Det bør brukes relative størrelser på fonter, marger og padding
- Ikke åpne nye vinduer uten å gjøre det klart for brukeren på forhånd
- Klargjør hvor linker peker

For at Altinn skal komme opp på nivå AAA må punktene under A og AA først tilfredsstilltes. Deretter må ytterligere 10 punkter fikses, jeg angir noen her, resten finnes på side 131:

- Legg inn fyll-tekst i tomme form-elementer
- Relaterte linker må grupperes
- Spesifiser forkortelser
- Angi språket i dokumentet
- Tilby sammendrag av tabeller
- Gruppér relaterte dokumenter / dokumentsamlinger



## 9. VIDERE ARBEID

### 9.1. Veien videre

Det er mye jeg ikke har hatt mulighet til å gjøre i mitt arbeid med denne oppgaven. Her vil jeg peke på en del ting jeg mener burde gjøres videre, basert på det jeg har gjort så langt.

#### Brukertesting og prototyping

Når det gjelder forslagene mine til forbedring av webportalen bør det gjøres en del testing og design av løsningsforslag for å videre raffinere og bedre forslagene. Mange av forslagene er ikke blitt testet på brukere – det burde det absolutt gjøres før man eventuelt går videre med forslagene for implementering. Mye av denne testingen kan gjøres i enkle prototyper, gjerne bygget i papir, eller som statiske skjermdumper.

For utforsking av universell utforming bør det bygges HTML-prototyper som viser at det er mulig å lage Altinn som tilfredsstiller kravene til i det minste A-standard. Sannsynligvis burde det være mulig å få til AA, men enkelte av disse kravene er vanskelig å teste uten at det bygges pilotsystemer. Grunnen til dette er den utstrakte bruken av javascript og annen dynamikk.

Videre bør det gjøres mer testing, spesielt sjekke flere skjemaer enn kun det ene skjemaet jeg har testet i denne oppgaven. Det bør også kjøres en full WCAG conformance test og full validering av sidene. Det bør også holdes noen brukertester for brukere som har funksjonshemninger, slik at man kan hente inn deres reaksjoner. Dersom dette ikke lar seg gjøre burde det i det *minste* la seg gjøre at noen utviklere eller testbrukere tester løsningen gjennom høyttalende nettlesere eller tekstbaserte nettlesere.

#### Realitetssjekk og kost/nytte

Jeg har prøvd å ikke tenke på hva som er, og ikke er mulig i dagens tekniske løsning når jeg har formulert løsningsforslagene i denne oppgaven. Tanken bak dette har vært at brukeren bør være den bestemmende faktoren i utformingen av brukergrensesnittet.

I virkeligheten er det ikke slik. Som designere må vi alle tilpasse oss begrensninger i form av teknologi, budsjett, tilgjengelig tid og ressurser osv. Likevel tror jeg det er nyttig å kunne komme med "optimale" løsningsforslag før vi binder kreativiteten med hva som er realistisk. Det er dette jeg har forsøkt å gjøre.

Før vi går videre med løsningsforslagene som er presentert må vi gjøre en "reality check". Hvilke av disse forslagene lar seg implementere, og hvor mye vil det koste?

Vil det være verdt ressursbruken å gjøre hele Altinn tilgjengelig? Kanskje det er urealistisk på nivå AAA, mens det lett lar seg å få til på nivå A.

Det ser ut til at skjemamotoren og den tilhørende skjemaeditoren har stor innvirkning på brukergrensesnittet i webportalen. Skjemamotoren legger begrensninger på både universell utforming og hva man kan gjøre med brukergrensesnittet ellers. Skjemamotoren og det tilhørende skjemabyggerverktøyet er en stor del av Altinns tekniske løsning. Å bytte ut, eller oppgradere, dette verktøyet vil kreve mye ressurser. Her må man se på hva som er mulig i nye versjoner, eventuelt i andre programvarepakker.

## 9.2. Hva vil fremtiden bringe?

Det er mye som skjer i offentlig sektor hva gjelder elektroniske tjenester og bruk av internett som plattform for å tilby disse tjenestene. Det er mange spennende prosjekter på gang, og for en som jobber med fokus på brukeropplevelser er det så mye spennende å ta tak i at det har vært vanskelig å plukke ut et passe stort stykke som jeg ville klare å bite over i denne oppgaven. Det følgende er tanker jeg har gjort meg under arbeidet med oppgaven, men som ikke ble med på grunn av omfang.

### Samordning og nye tjenester

Til nå har de fleste offentlige tjenester som har blitt lansert på nettet vært separate "øyer". Med unntak av Altinn har de fleste etater jobbet med sine egne ting og fokusert på sine egne tjenester. Norge.no har vært et forsøk på å veilede borgerne fram til informasjon, men det har vært liten reell samordning. Altinn var et første skritt for å samle tjenester på ett sted.

Morten Meyers visjon om "Min side", en borgerportal under Norge.no som skal samle statlige- og kommunale tjenester og informasjon på ett sted, er et nytt skritt på veien.

Disse nye tankene og tjenestene fører til mange spørsmål. Det første spørsmålet er hvordan disse tjenestene skal organiseres i forhold til hverandre, og hvordan de skal utfylle hverandre. En bruker står potensielt ovenfor tre nettsider dersom han vil gjøre endringer eller finne informasjon av skattemessig art:

- 1) Skatteetaten.no
- 2) Altinn.no
- 3) Norge.no / min side

Hvordan skal disse tjenestene organiseres slik at både brukere og forvaltning får en best mulig løsning? Hva skal legges hvor? – Her vil det kreves et overordnet blikk og en strategi som går langt over hva det offentlige har tenkt hittil.

Med disse nye tjenestene og portalene er en ting i hvert fall sikkert. Det er ikke lenger slik at etatene kun kan forholde seg til sitt eget nettsted og sin egen tjeneste.

*Fremtiden er et nettverk av nettsider.*

Et nettverk av offentlige nettsteder bringer med seg spørsmål om såkalt "common look and feel" – et sett med retningslinjer for hvordan offentlige nettsteder er bygget opp og ser ut. Canada har det. Hong Kong har det. Norge *bør* ha det. Tanken er at dette skal lette navigering og bruk for borgerne. Dette er ikke ulikt tanken bak ELMER-retningslinjene for elektroniske skjema. Det er nødvendig å dra i gang et liknende arbeid for offentlige nettsteder.

### **Informasjonsbehovet burde være utgangspunktet for skjema**

I dag er de fleste store etatene satt opp for å lage papirskjema. De elektroniske skjemaene som lages i dag er for det meste noe som lages ved siden av – som regel basert på papirskjemaet. Hele skjemaprosessen er sentrert rundt det å lage papirskjema, hvilket betyr at produksjonsprosesser er optimalisert for papir. Hjelpetekster skrives for papir, utgangspunktet for et elektronisk skjema er papirutgaven.

Dette blir feil utgangspunkt for de elektroniske skjemaene. I stedet for å basere seg på papirskjemaet som en slags fasit burde etatene se på *informasjonsbehovet*, uavhengig av mediet. Det er informasjonsbehovet som burde stå i sentrum. Ut i fra dette behovet bør det være mulig å hente inn denne informasjonen i forskjellige kanaler og medier. I dag opererer man stort sett med papirskjema, fagsystemer og webskjema. Men det kan godt tenkes at det kommer flere kanaler – for eksempel mobilskjema. For å kunne møte denne utfordringen på en bedre måte bør skjemaprosessene i etatene bygges om. Dette er en videreføring av arbeidet med elektroniske skjema og vil kreve langsiktige tanker og strategisk arbeid.

### **9.3. Til slutt**

Når det gjelder utformingen av skjema i Altinn er Skatteetaten allerede i gang med gjennomføre endringer i sine skjema. Hvor mye ressurser som settes inn vil være avgjørende på hvor vellykket arbeidet vil bli. Retningslinjene er på plass, men retningslinjer alene er ikke nok for å få gode elektroniske skjema. Den store utfordringen i å lage gode skjema på nettet er at det krever kompetanse fra mange fagfelt.

Det skjer mye spennende arbeid på e-skjema-fronten om dagen. Samtidig forbedres stadig Altinn, og om ikke alt for lenge skal Min side lanseres for fullt.

Jeg håper denne oppgaven har bidratt med noen nyttige tanker om mulige forbedringer i Altinn. Jeg kommer til å følge utviklingen av Altinn med spenning!



## REFERANSER

AAD (2001). Strategi og tiltak for døgnåpen forvaltning. Arbeids- og administrasjonsdepartementet.

Accenture (2004a). Brukerdokumentasjon for skjema utvikling (v5.1). Altinn dokumentasjon.

Accenture (2004b). Standard for utforming av brukergrensesnitt (v5.0). Altinn dokumentasjon.

Bowman, D. (2004). Throwing Tables Out The Window.  
[http://www.stopdesign.com/articles/throwing\\_tables/](http://www.stopdesign.com/articles/throwing_tables/) (besøkt 28-08-2004)

Christensen, B. H. (2005). Arkitekter, arkitekturer og arkitekturplanlegging (kursdokumentasjon, Software 2005).

Dieberger, A. (1994). Navigation in Textual Virtual Environments using a City Metaphor, Vienna University of Technology.

DWP (1995). Disability Discrimination Act.  
[www.disability.gov.uk/dda/](http://www.disability.gov.uk/dda/) (besøkt 30-9-2004)

Ernes, A. K. B. (2004a). Elektronisk rapportering fungerer ikke. Digi.no 12.05.2004  
<http://www.digi.no/php/art.php?id=104031>

Ernes, A. K. B. (2004b). Treg overgang til digital forvaltning. Digi.no 25.10.2004  
<http://www.digi.no/php/art.php?id=111328>

Festa, P. (2003). CNET news.com: Developers gripe about IE standards inaction.  
[http://news.com.com/2100-1032\\_3-5088642.html](http://news.com.com/2100-1032_3-5088642.html) (besøkt 28.07.2005)

Garrett, J. J. (2003). The Elements of User Experience: User-Centered Design for the Web, New Riders.

Garrett, J. J. (2005). Ajax: A New Approach to Web Applications.  
<http://www.adaptivepath.com/publications/essays/archives/000385.php>  
(besøkt 30.07.2005)

Haraldsen, G. (2004). Råd for utvikling og utforming av webskjema (v1.1). Statistisk sentralbyrås håndbøker, Statistisk sentralbyrå.

iCan (2004). iCan – The Leading Disability Community.  
<http://www.icanonline.net/> (besøkt 18-10-2004)

Jahr, C. og Tønnesen, E. (2004a). Hudfletter Altinn-prosjektet. Computerworld 18.06.2004  
<http://www.computerworld.no/index.cfm?fuseaction=artikkel&id=3260FF1D-B4C7-5FD1-273281DF43D2FE8E>

Jahr, C. og Tønnesen, E. (2004b). -- Løsningen blir stadig bedre. Computerworld 18.06.2004  
[http://computerworld.no/index.cfm/ill\\_liste/artikkel/id/43211](http://computerworld.no/index.cfm/ill_liste/artikkel/id/43211)

- Jahr, C. og Tønnesen, E. (2004c). -- Viktigere å holde tidsfristen. Computerworld 18.06.2004 [http://computerworld.no/index.cfm/ill\\_liste/artikkel/id/43207](http://computerworld.no/index.cfm/ill_liste/artikkel/id/43207)
- Krug, S. (2000). Don't Make Me Think! A Common Sense Approach to Web Usability, New Riders.
- MinSide-prosjektet (2004). Nordisk portalmøte 2004: Min side – en enklere hverdag for deg og meg.
- Morville, P. (2004a). Findability.org – The Design of Findable Objects.  
<http://findability.org/faq.php> (besøkt 23.04.2005)
- Morville, P. (2004b). User Experience Design.  
<http://semanticstudios.com/publications/semantics/000029.php> (besøkt 14.03.2005)
- Moss, T. (2004). Secret Benefits of Accessibility Part 1: Increased Usability.  
<http://www.sitepoint.com/article/accessible-usable-website> (besøkt 15-10-2004)
- NHD (2002). e-Norge 2005.
- Nielsen, J. (19?). Severity Ratings for Usability Problems.  
<http://www.useit.com/papers/heuristic/severityrating.html> (besøkt 23.05.2005)
- Nielsen, J. (1999a). Designing Web Usability: The Practice of Simplicity, New Riders.
- Nielsen, J. (1999b). The Top Ten New Mistakes of Web Design.  
<http://www.useit.com/alertbox/990530.html> (besøkt 26.07.2005)
- Nielsen, J. (2002). Top Ten Web-Design Mistakes of 2002.  
<http://www.useit.com/alertbox/20021223.html> (besøkt 29.07.2005)
- Nielsen, J. (2003). Information Foraging: Why Google Makes People Leave Your Site Faster.  
<http://www.useit.com/alertbox/20030630.html> (besøkt 25.07.2005)
- Nielsen, J. (2005). Scrolling and Scrollbars.  
<http://www.useit.com/alertbox/20050711.html> (besøkt 29.07.2005)
- Nielsen, J. og Mack, R. L. (1994). Usability Inspection Methods, John Wiley & Sons.
- Norman, D. A. (2002). The Design of Everyday Things, Basic Books.
- Nygaard, T. (2000). Behandling av offentlige skjemaer i utvalgte bedrifter. ELMER, Nærings- og Handelsdepartementet.
- Nygaard, T. (2001a). Bedriftene og skjemaveldet – Observasjoner etter ett års planlegging. ELMER, Nærings- og handelsdepartementet.
- Nygaard, T. (2001b). Skjemaet som lærte å lytte – Et eksempel på elektronisk dialog. ELMER, Nærings- og Handelsdepartementet.
- Nygaard, T. (2003). IT mot skjemabelastning – Elektronisk rapportering som forenklingstiltak for næringslivet. ELMER, Nærings- og Handelsdepartementet.

- Pirolli, P. og Card, S. K. (1999). "Information Foraging." UIR Technical Report.
- Preece, J., Rogers, Y., et al. (2002). Interaction Design: Beyond Human-Computer Interaction, John Wiley & Sons.
- Rosenfeld, L. og Morville, P. (2002). Information Architecture for the World Wide Web.
- Shneiderman, B. og Plaisant, C. (2005). Designing the User Interface, Addison-Wesley.
- Skattedirektoratet (2002). Avtale om utvikling av programvare – Utvikling av Altinn (1/6), Skattedirektoratet.
- Skattedirektoratet (2005a). Rekordmange leverte selvangivelsen elektronisk. <http://skatteetaten.no/Templates/Pressemelding.aspx?id=28875> (besøkt 04.05.2005)
- Skattedirektoratet (2005b). Retningslinjer for elektroniske skjemaer i Skattedirektoratet (v1.2), Skattedirektoratet.
- Skatteetaten (2004). Strategier 2005-2008. Skatteetaten.
- Sosialdepartementet (2002-2003). St.meld. nr. 40: Nedbygging av funksjonshemmende barrierer.
- SSB (2002). Levekårsundersøkelsen.
- Statskonsult (2004). Kvalitetskriterier for offentlige nettsteder. [www.kvalitetpaanett.net](http://www.kvalitetpaanett.net) (besøkt 07-11-2004)
- Tognazzini, B. (2003). First Principles of Interaction Design. <http://www.asktog.com/basics/firstPrinciples.html> (besøkt 12.04.2005)
- Tønnesen, E. (2004a). Altinn - og ingenting ut. Computerworld 11.06.2004 [http://computerworld.no/index.cfm/ill\\_liste/artikkel/id/43034](http://computerworld.no/index.cfm/ill_liste/artikkel/id/43034)
- Tønnesen, E. (2004b). De små misfornøyd med Altinn. Computerworld 08.12.2004 [http://computerworld.no/index.cfm/ill\\_liste/artikkel/id/47405](http://computerworld.no/index.cfm/ill_liste/artikkel/id/47405)
- Tønnesen, E. (2004c). Mye mer jobb med Altinn. Computerworld 14.06.2004 <http://computerworld.no/index.cfm/fuseaction/artikkel/id/43029>
- Tønnesen, E. (2004d). -- Stort forbederingspotensial. Computerworld 14.06.2004 <http://computerworld.no/index.cfm/fuseaction/artikkel/id/43028>
- Udjus, L. (2005). Erfaringer med SKD-skjema i Altinn (intern rapport), Skattedirektoratet.
- USC (1998). Section 508. <http://www.section508.gov/> (besøkt 01-11-2004)
- W3C (1999-2004). World Wide Web Consortium (W3C). [www.w3.org](http://www.w3.org) (besøkt 01-11-2004)

- W3C/WAI (1999). Web Content Accessibility Guidelines 1.0.  
<http://www.w3.org/TR/WAI-WEBCONTENT/> (besøkt 01-10-2004)
- W3C/WAI (2003). Online Overview of the Web Accessibility Initiative.  
<http://www.w3.org/Talks/WAI-Intro/Overview.html> (besøkt 26.07.2005)
- WASP (2002). Web Standards Project.  
[www.webstandards.org](http://www.webstandards.org) (besøkt 28.07.2005)
- Watchfire (2004). Watchfire.com.  
[www.watchfire.com](http://www.watchfire.com) (besøkt 01.11.2004)